CS 4110

Programming Languages & Logics

Victory Lap!

Victory Lap

We've covered a *lot* of territory this semester…

…now you're ready to soar!

Semester in Review

CS 4110 Home Project Resources Schedule Syllabus CMS Campuswire

Schedule

Date	Topic	Notes	
September 2	Course Overview	slides video	
September 4	Introduction to Semantics	slides notes video	
September 7	Inductive Definitions	slides notes video	
September 9	Properties and Inductive Proofs	same slides same notes video	
September 11	Inductive Proof and Large-Step Semantics	slides notes video	
September 14	The IMP Language	slides notes video	
September 16	IMP Properties	slides notes video	
September 18	More IMP Proofs	same slides same notes whiteboard-1 whiteboard-2 video-1 video-2	
September 21	Denotational Semantics	slides notes video	
September 23	Denotational Semantics Examples	slides notes video	
September 25	Axiomatic Semantics	slides notes video	
September 28	Hoare Logic	slides notes video	
September 30	Hoare Logic Examples	slides notes video	
October 2	Weakest Preconditions	slides notes video	
October 5	λ-Calculus	notes video	
October 7	More λ-Calculus and Substitution	slides notes video	
October 9	de Bruijn and Combinators	slides notes video	
October 12	Encodings	slides notes video	
October 14	**Mid-semester break**		
October 16	Encodings and Fixed-Point Combinators	slides notes video	
October 19	Definitional Translation	slides notes video	HW4 due
October 21	Continuations	slides notes video	
October 23	Types	slides notes video	
October 26	More Types	same slides same notes video	HW5 due
October 28	Proving Type Soundness	slides notes video	
October 30	More Proving Type Soundness	same slides same notes video	
November 2	Normalization	slides notes video	HW6 due
November 4	Advanced Types	slides notes video	
November 6	Polymorphism	slides notes video	
November 9	Type Inference	slides notes video	Take-Home Prelim 2
November 11	Recursive Types	slides notes video	
November 13	Records and Subtyping	slides notes video	Project Alpha due
November 16	**Study days**		
November 18	**Semi-finals**		
November 20	**Semi-finals**		
November 23	**Semi-finals**		
November 25	**No class (Thanksgiving)**		
November 27	**No class (Thanksgiving)**		
November 30	Existential Types	slides notes video	
December 2	Propositions as Types	slides notes video	
December 4	Dependent Types	notes video code	Project Beta due
December 7	Concurrency	slides video	Homework 7 due
December 9	Probabilistic Semantics		
December 11	After 4110	slides	
December 14	DSLs and Bidirectional Programming	slides video	Homework 8 due
December 16	Victory Lap		Final Project due

Semester in Review

CS 4110 Home Project Resources Schedule Syllabus CMS Campuswire

Schedule

Mathematical Preliminaries

Date	Topic	Notes	
September 2	Course Overview	slides video	
September 4	Introduction to Semantics	slides notes video	
September 7	Inductive Definitions	slides notes video	
September 9	Properties and Inductive Proofs	same slides same notes video	
September 11	Inductive Proof and Large-Step Semantics	slides notes video	
September 14	The IMP Language	slides notes video	
September 16	IMP Properties	slides notes video	
September 18	More IMP Proofs	same slides same notes whiteboard-1 whiteboard-2 video-1 video-2	
September 21	Denotational Semantics	slides notes video	
September 23	Denotational Semantics Examples	slides notes video	
September 25	Axiomatic Semantics	slides notes video	
September 28	Hoare Logic	slides notes video	
September 30	Hoare Logic Examples	slides notes video	
October 2	Weakest Preconditions	slides notes video	
October 5	λ-Calculus	notes video	
October 7	More λ-Calculus and Substitution	slides notes video	
October 9	de Bruijn and Combinators	slides notes video	
October 12	Encodings	slides notes video	
October 14	**Mid-semester break**		
October 16	Encodings and Fixed-Point Combinators	slides notes video	
October 19	Definitional Translation	slides notes video	HW4 due
October 21	Continuations	slides notes video	
October 23	Types	slides notes video	
October 26	More Types	same slides same notes video	HW5 due
October 28	Proving Type Soundness	slides notes video	
October 30	More Proving Type Soundness	same slides same notes video	
November 2	Normalization	slides notes video	HW6 due
November 4	Advanced Types	slides notes video	
November 6	Polymorphism	slides notes video	
November 9	Type Inference	slides notes video	Take-Home Prelim 2
November 11	Recursive Types	slides notes video	
November 13	Records and Subtyping	slides notes video	Project Alpha due
November 16	**Study days**		
November 18	**Semi-finals**		
November 20	**Semi-finals**		
November 23	**Semi-finals**		
November 25	**No class (Thanksgiving)**		
November 27	**No class (Thanksgiving)**		
November 30	Existential Types	slides notes video	
December 2	Propositions as Types	slides notes video	
December 4	Dependent Types	notes video code	Project Beta due
December 7	Concurrency	slides video	Homework 7 due
December 9	Probabilistic Semantics		
December 11	After 4110	slides	
December 14	DSLs and Bidirectional Programming	slides video	Homework 8 due
December 16	Victory Lap		Final Project due

Semester in Review

CS 4110 Home Project Resources Schedule Syllabus CMS Campuswire

Schedule

Mathematical Preliminaries

Operational Semantics

Date	Topic	Notes	
September 2	Course Overview	slides video	
September 4	Introduction to Semantics	slides notes video	
September 7	Inductive Definitions	slides notes video	
September 9	Properties and Inductive Proofs	same slides same notes video	
September 11	Inductive Proof and Large-Step Semantics	slides notes video	
September 14	The IMP Language	slides notes video	
September 16	IMP Properties	slides notes video	
September 18	More IMP Proofs	same slides same notes whiteboard-1 whiteboard-2 video-1 video-2	
September 21	Denotational Semantics	slides notes video	
September 23	Denotational Semantics Examples	slides notes video	
September 25	Axiomatic Semantics	slides notes video	
September 28	Hoare Logic	slides notes video	
September 30	Hoare Logic Examples	slides notes video	
October 2	Weakest Preconditions	slides notes video	
October 5	λ-Calculus	notes video	
October 7	More λ-Calculus and Substitution	slides notes video	
October 9	de Bruijn and Combinators	slides notes video	
October 12	Encodings	slides notes video	
October 14	**Mid-semester break**		
October 16	Encodings and Fixed-Point Combinators	slides notes video	
October 19	Definitional Translation	slides notes video	HW4 due
October 21	Continuations	slides notes video	
October 23	Types	slides notes video	
October 26	More Types	same slides same notes video	HW5 due
October 28	Proving Type Soundness	slides notes video	
October 30	More Proving Type Soundness	same slides same notes video	
November 2	Normalization	slides notes video	HW6 due
November 4	Advanced Types	slides notes video	
November 6	Polymorphism	slides notes video	
November 9	Type Inference	slides notes video	Take-Home Prelim 2
November 11	Recursive Types	slides notes video	
November 13	Records and Subtyping	slides notes video	Project Alpha due
November 16	**Study days**		
November 18	**Semi-finals**		
November 20	**Semi-finals**		
November 23	**Semi-finals**		
November 25	**No class (Thanksgiving)**		
November 27	**No class (Thanksgiving)**		
November 30	Existential Types	slides notes video	
December 2	Propositions as Types	slides notes video	
December 4	Dependent Types	notes video code	Project Beta due
December 7	Concurrency	slides video	Homework 7 due
December 9	Probabilistic Semantics		
December 11	After 4110	slides	
December 14	DSLs and Bidirectional Programming	slides video	Homework 8 due
December 16	Victory Lap		Final Project due

Semester in Review

CS 4110 Home Project Resources Schedule Syllabus CMS Campuswire

Schedule

Mathematical Preliminaries

Date	Topic	Notes
September 2	Course Overview	slides video
September 4	Introduction to Semantics	slides notes video
September 7	Inductive Definitions	slides notes video
September 9	Properties and Inductive Proofs	same slides same notes video

Operational Semantics

Date	Topic	Notes
September 11	Inductive Proof and Large-Step Semantics	slides notes video
September 14	The IMP Language	slides notes video
September 16	IMP Properties	slides notes video
September 18	More IMP Proofs	same slides same notes whiteboard-1 whiteboard-2 video-1 video-2

Denotational Semantics

Date	Topic	Notes
September 21	Denotational Semantics	slides notes video
September 23	Denotational Semantics	slides notes video
September 25	Axiomatic Semantics	slides notes video

Date	Topic	Notes	
September 28	Hoare Logic	slides notes video	
September 30	Hoare Logic Examples	slides notes video	
October 2	Weakest Preconditions	slides notes video	
October 5	λ-Calculus	notes video	
October 7	More λ-Calculus and Substitution	slides notes video	
October 9	de Bruijn and Combinators	slides notes video	
October 12	Encodings	slides notes video	
October 14	**Mid-semester break**		
October 16	Encodings and Fixed-Point Combinators	slides notes video	
October 19	Definitional Translation	slides notes video	HW4 due
October 21	Continuations	slides notes video	
October 23	Types	slides notes video	
October 26	More Types	same slides same notes video	HW5 due
October 28	Proving Type Soundness	slides notes video	
October 30	More Proving Type Soundness	same slides same notes video	
November 2	Normalization	slides notes video	HW6 due
November 4	Advanced Types	slides notes video	
November 6	Polymorphism	slides notes video	
November 9	Type Inference	slides notes video	Take-Home Prelim 2
November 11	Recursive Types	slides notes video	
November 13	Records and Subtyping	slides notes video	Project Alpha due
November 16	**Study days**		
November 18	**Semi-finals**		
November 20	**Semi-finals**		
November 23	**Semi-finals**		
November 25	**No class (Thanksgiving)**		
November 27	**No class (Thanksgiving)**		
November 30	Existential Types	slides notes video	
December 2	Propositions as Types	slides notes video	
December 4	Dependent Types	notes video code	Project Beta due
December 7	Concurrency	slides video	Homework 7 due
December 9	Probabilistic Semantics		
December 11	After 4110	slides	
December 14	DSLs and Bidirectional Programming	slides video	Homework 8 due
December 16	Victory Lap		Final Project due

Semester in Review

CS 4110 Home Project Resources Schedule Syllabus CMS Campuswire

Schedule

Date	Topic	Notes	
	Mathematical Preliminaries		
September 2	Course Overview	slides video	
September 4	Introduction to Semantics	slides notes video	
September 7	Inductive Definitions	slides notes video	
September 9	Properties and Inductive Proofs	same slides same notes video	
	Operational Semantics		
September 11	Inductive Proof and Large-Step Semantics	slides notes video	
September 14	The IMP Language	slides notes video	
September 16	IMP Properties	slides notes video	
September 18	More IMP Proofs	same slides same notes whiteboard-1 whiteboard-2 video-1 video-2	
	Denotational Semantics		
September 21	Denotational Semantics	slides notes video	
September 23	Denotational Semantics	slides notes video	
September 25	Axiomatic Semantics	slides notes video	
	Axiomatic Semantics		
September 28	Hoare Logic	slides notes video	
September 30	Hoare Logic Examples	slides notes video	
October 2	Weakest Preconditions	slides notes video	
October 5	λ-Calculus	notes video	
October 7	More λ-Calculus and Substitution	slides notes video	
October 9	de Bruijn and Combinators	slides notes video	
October 12	Encodings	slides notes video	
October 14	**Mid-semester break**		
October 16	Encodings and Fixed-Point Combinators	slides notes video	
October 19	Definitional Translation	slides notes video	HW4 due
October 21	Continuations	slides notes video	
October 23	Types	slides notes video	
October 26	More Types	same slides same notes video	HW5 due
October 28	Proving Type Soundness	slides notes video	
October 30	More Proving Type Soundness	same slides same notes video	
November 2	Normalization	slides notes video	HW6 due
November 4	Advanced Types	slides notes video	
November 6	Polymorphism	slides notes video	
November 9	Type Inference	slides notes video	Take-Home Prelim 2
November 11	Recursive Types	slides notes video	
November 13	Records and Subtyping	slides notes video	Project Alpha due
November 16	**Study days**		
November 18	**Semi-finals**		
November 20	**Semi-finals**		
November 23	**Semi-finals**		
November 25	**No class (Thanksgiving)**		
November 27	**No class (Thanksgiving)**		
November 30	Existential Types	slides notes video	
December 2	Propositions as Types	slides notes video	
December 4	Dependent Types	notes video code	Project Beta due
December 7	Concurrency	slides video	Homework 7 due
December 9	Probabilistic Semantics		
December 11	After 4110	slides	
December 14	DSLs and Bidirectional Programming	slides video	Homework 8 due
December 16	Victory Lap		Final Project due

Semester in Review

CS 4110 Home Project Resources Schedule Syllabus CMS Campuswire

Schedule

Date	Topic	Notes
	Mathematical Preliminaries	
September 2	Course Overview	slides video
September 4	Introduction to Semantics	slides notes video
September 7	Inductive Definitions	slides notes video
September 9	Properties and Inductive Proofs	same slides same notes video
	Operational Semantics	
September 11	Inductive Proof and Large-Step Semantics	slides notes video
September 14	The IMP Language	slides notes video
September 16	IMP Properties	slides notes video
September 18	More IMP Proofs	same slides same notes whiteboard-1 whiteboard-2 video-1 video-2
	Denotational Semantics	
September 21	Denotational Semantics	slides notes video
September 23	Denotational Se...	slides notes video
September 25	Axiomatic Semantics	slides notes video
	Axiomatic Semantics	
September 28	Hoare Logic	slides notes video
September 30	Hoare Logic Examp...	slides notes video
October 2	Weakest Preconditions	slides notes video
	λ-Calculus	
October 5	λ-Calculus	notes video
October 7	More λ-Calculus and Substitution	slides notes video
October 9	de Bruijn and Combinators	slides notes video
October 12	Encodings	slides notes video
October 14	**Mid-semester break**	

October 16	Encodings and Fixed-Point Combinators	slides notes video	
October 19	Definitional Translation	slides notes video	HW4 due
October 21	Continuations	slides notes video	
October 23	Types	slides notes video	
October 26	More Types	same slides same notes video	HW5 due
October 28	Proving Type Soundness	slides notes video	
October 30	More Proving Type Soundness	same slides same notes video	
November 2	Normalization	slides notes video	HW6 due
November 4	Advanced Types	slides notes video	
November 6	Polymorphism	slides notes video	
November 9	Type Inference	slides notes video	Take-Home Prelim 2
November 11	Recursive Types	slides notes video	
November 13	Records and Subtyping	slides notes video	Project Alpha due
November 16	**Study days**		
November 18	**Semi-finals**		
November 20	**Semi-finals**		
November 23	**Semi-finals**		
November 25	**No class (Thanksgiving)**		
November 27	**No class (Thanksgiving)**		
November 30	Existential Types	slides notes video	
December 2	Propositions as Types	slides notes video	
December 4	Dependent Types	notes video code	Project Beta due
December 7	Concurrency	slides video	Homework 7 due
December 9	Probabilistic Semantics		
December 11	After 4110	slides	
December 14	DSLs and Bidirectional Programming	slides video	Homework 8 due
December 16	Victory Lap		Final Project due

Semester in Review

CS 4110 Home Project Resources Schedule Syllabus CMS Campuswire

Schedule

Mathematical Preliminaries

Date	Topic	Notes
September 2	Course Overview	slides video
September 4	Introduction to Semantics	slides notes video
September 7	Inductive Definitions	slides notes video
September 9	Properties and Inductive Proofs	same slides same notes video

Operational Semantics

Date	Topic	Notes
September 11	Inductive Proof and Large-Step Semantics	slides notes video
September 14	The IMP Language	slides notes video
September 16	IMP Properties	slides notes video
September 18	More IMP Proofs	same slides same notes whiteboard-1 whiteboard-2 video-1 video-2

Denotational Semantics

Date	Topic	Notes
September 21	Denotational Semantics	slides notes video
September 23	Denotational Semantics	slides notes video
September 25	Axiomatic Semantics	slides notes video

Axiomatic Semantics

Date	Topic	Notes
September 28	Hoare Logic	slides notes video
September 30	Hoare Logic Examples	slides notes video
October 2	Weakest Preconditions	slides notes video

λ-Calculus

Date	Topic	Notes
October 5	λ-Calculus	notes video
October 7	More λ-Calculus and Substitution	slides notes video
October 9	de Bruijn and Combinators	slides notes video
October 12	Encodings	slides notes video
October 14	**Mid-semester break**	

λ-Calculus Programming

Date	Topic	Notes	Due
October 16	Encodings and Fixed-Point Combinators	slides notes video	
October 19	Definitional Translation	slides notes video	HW4 due
October 21	Continuations	slides notes video	

Date	Topic	Notes	Due
October 23	Types	slides notes video	
October 26	More Types	same slides same notes video	HW5 due
October 28	Proving Type Soundness	slides notes video	
October 30	More Proving Type Soundness	same slides same notes video	
November 2	Normalization	slides notes video	HW6 due
November 4	Advanced Types	slides notes video	
November 6	Polymorphism	slides notes video	
November 9	Type Inference	slides notes video	Take-Home Prelim 2
November 11	Recursive Types	slides notes video	
November 13	Records and Subtyping	slides notes video	Project Alpha due
November 16	**Study days**		
November 18	**Semi-finals**		
November 20	**Semi-finals**		
November 23	**Semi-finals**		
November 25	**No class (Thanksgiving)**		
November 27	**No class (Thanksgiving)**		
November 30	Existential Types	slides notes video	
December 2	Propositions as Types	slides notes video	
December 4	Dependent Types	notes video code	Project Beta due
December 7	Concurrency	slides video	Homework 7 due
December 9	Probabilistic Semantics		
December 11	After 4110	slides	
December 14	DSLs and Bidirectional Programming	slides video	Homework 8 due
December 16	Victory Lap		Final Project due

Semester in Review

CS 4110 Home Project Resources Schedule Syllabus CMS Campuswire

Schedule

Mathematical Preliminaries

Date	Topic	Notes	
September 2	Course Overview	slides video	
September 4	Introduction to Semantics	slides notes video	
September 7	Inductive Definitions	slides notes video	
September 9	Properties and Inductive Proofs	same slides same notes video	

Operational Semantics

Date	Topic	Notes	
September 11	Inductive Proof and Large-Step Semantics	slides notes video	
September 14	The IMP Language	slides notes video	
September 16	IMP Properties	slides notes video	
September 18	More IMP Proofs	same slides same notes whiteboard-1 whiteboard-2 video-1 video-2	

Denotational Semantics

Date	Topic	Notes	
September 21	Denotational Semantics	slides notes video	
September 23	Denotational Semantics Examples	slides notes video	
September 25	Axiomatic Semantics	slides notes video	

Axiomatic Semantics

Date	Topic	Notes	
September 28	Hoare Logic	slides notes video	
September 30	Hoare Logic Examples	slides notes video	
October 2	Weakest Preconditions	slides notes video	

λ-Calculus

Date	Topic	Notes	
October 5	λ-Calculus	notes video	
October 7	More λ-Calculus and Substitution	slides notes video	
October 9	de Bruijn and Combinators	slides notes video	
October 12	Encodings	slides notes video	
October 14	**Mid-semester break**		

λ-Calculus Programming

Date	Topic	Notes	
October 16	Encodings and Fixed-Point Combinators	slides notes video	
October 19	Definitional Translation	slides notes video	HW4 due
October 21	Continuations	slides notes video	

Simple Types

Date	Topic	Notes	
October 23	Types	slides notes video	
October 26	More Types	same slides same notes video	HW5 due
October 28	Proving Type Soundness	slides notes video	
October 30	More Proving Type Soundness	same slides same notes video	
November 2	Normalization	slides notes video	HW6 due

Date	Topic	Notes	
November 4	Advanced Types	slides notes video	
November 6	Polymorphism	slides notes video	
November 9	Type Inference	slides notes video	Take-Home Prelim 2
November 11	Recursive Types	slides notes video	
November 13	Records and Subtyping	slides notes video	Project Alpha due
November 16	**Study days**		
November 18	**Semi-finals**		
November 20	**Semi-finals**		
November 23	**Semi-finals**		
November 25	**No class (Thanksgiving)**		
November 27	**No class (Thanksgiving)**		
November 30	Existential Types	slides notes video	
December 2	Propositions as Types	slides notes video	
December 4	Dependent Types	notes video code	Project Beta due
December 7	Concurrency	slides video	Homework 7 due
December 9	Probabilistic Semantics		
December 11	After 4110	slides	
December 14	DSLs and Bidirectional Programming	slides video	Homework 8 due
December 16	Victory Lap		Final Project due

Semester in Review

CS 4110 Home Project Resources Schedule Syllabus CMS Campuswire

Schedule

Mathematical Preliminaries

Date	Topic	Notes	
September 2	Course Overview	slides video	
September 4	Introduction to Semantics	slides notes video	
September 7	Inductive Definitions	slides notes video	
September 9	Properties and Inductive Proofs	same slides same notes video	

Operational Semantics

Date	Topic	Notes	
September 11	Inductive Proof and Large-Step Semantics	slides notes video	
September 14	The IMP Language	slides notes video	
September 16	IMP Properties	slides notes video	
September 18	More IMP Proofs	same slides same notes whiteboard-1 whiteboard-2 video-1 video-2	

Denotational Semantics

Date	Topic	Notes	
September 21	Denotational Semantics	slides notes video	
September 23	Denotational [illegible]	[illegible]	
September 25	Axiomatic Semantics	slides notes video	

Axiomatic Semantics

Date	Topic	Notes	
September 28	Hoare Logic	slides notes video	
September 30	Hoare Logic Examples	[illegible]	
October 2	Weakest Preconditions	slides notes video	

λ-Calculus

Date	Topic	Notes	
October 5	λ-Calculus	notes video	
October 7	More λ-Calculus and Substitution	slides notes video	
October 9	de Bruijn and Combinators	[illegible]	
October 12	Encodings	slides notes video	
October 14	**Mid-semester break**		

λ-Calculus Programming

Date	Topic	Notes	
October 16	Encodings and Fixed-Point Combinators	slides notes video	
October 19	Definitional Translation	slides notes video	HW4 due
October 21	Continuations	slides notes video	

Simple Types

Date	Topic	Notes	
October 23	Types	slides notes video	
October 26	More Types	same slides same notes video	HW5 due
October 28	Proving Type Soundness	[illegible]	
October 30	More Proving Type Soundness	same slides same notes video	
November 2	Normalization	slides notes video	HW6 due

Advanced Types

Date	Topic	Notes	
November 4	Advanced Types	slides notes video	
November 6	Polymorphism	slides notes video	
November 9	Type Inference	[illegible]	Take-Home Prelim 2
November 11	Recursive Types	slides notes video	
November 13	Records and Subtyping	slides notes video	Project Alpha due

Date	Topic	Notes	
November 16	**Study days**		
November 18	**Semi-finals**		
November 20	**Semi-finals**		
November 23	**Semi-finals**		
November 25	**No class (Thanksgiving)**		
November 27	**No class (Thanksgiving)**		
November 30	Existential Types	slides notes video	
December 2	Propositions as Types	slides notes video	
December 4	Dependent Types	notes video code	Project Beta due
December 7	Concurrency	slides video	Homework 7 due
December 9	Probabilistic Semantics		
December 11	After 4110	slides	
December 14	DSLs and Bidirectional Programming	slides video	Homework 8 due
December 16	Victory Lap		Final Project due

Semester in Review

CS 4110 Home Project Resources Schedule Syllabus CMS Campuswire

Schedule

Mathematical Preliminaries

Operational Semantics

Denotational Semantics

Axiomatic Semantics

λ-Calculus

Date	Topic	Notes
September 2	Course Overview	slides video
September 4	Introduction to Semantics	
September 7	Inductive Definitions	slides notes video
September 9	Properties and Inductive Proofs	same slides same notes video
September 11	Inductive Proof and Large-Step Semantics	slides notes video
September 14	The IMP Language	
September 16	IMP Properties	slides notes video
September 18	More IMP Proofs	same slides same notes whiteboard-1 whiteboard-2 video-1 video-2
September 21	Denotational Semantics	slides notes video
September 23	Denotational Semantics	
September 25	Axiomatic Semantics	slides notes video
September 28	Hoare Logic	slides notes video
September 30	Hoare Logic Examples	
October 2	Weakest Preconditions	slides notes video
October 5	λ-Calculus	notes video
October 7	More λ-Calculus and Substitution	
October 9	de Bruijn and Combinators	
October 12	Encodings	slides notes video
October 14	**Mid-semester break**	

λ-Calculus Programming

Simple Types

Advanced Types

Logic and Proof

October 16	Encodings and Fixed-Point Combinators	slides notes video	
October 19	Definitional Translation		HW4 due
October 21	Continuations	slides notes video	
October 23	Types	slides notes video	
October 26	More Types	same slides same notes video	HW5 due
October 28	Proving Type Soundness		
October 30	More Proving Type Soundness	same slides same notes video	
November 2	Normalization	slides notes video	HW6 due
November 4	Advanced Types	slides notes video	
November 6	Polymorphism	slides notes video	
November 9	Type Inference		Take-Home Prelim 2
November 11	Recursive Types	slides notes video	
November 13	Records and Subtyping	slides notes video	Project Alpha due
November 16	**Study days**		
November 18	**Semi-finals**		
November 20	**Semi-finals**		
November 23	**Semi-finals**		
November 25	**No class (Thanksgiving)**		
November 27	**No class (Thanksgiving)**		
November 30	Existential Types	slides notes video	
December 2	Propositions as Types		
December 4	Dependent Types		Project Beta due
December 7	Concurrency	slides video	Homework 7 due
December 9	Probabilistic Semantics		
December 11	After 4110	slides	
December 14	DSLs and Bidirectional Programming	slides video	Homework 8 due
December 16	Victory Lap		Final Project due

Semester in Review

CS 4110 Home Project Resources Schedule Syllabus CMS Campuswire

Schedule

Mathematical Preliminaries

Date	Topic	Notes
September 2	Course Overview	slides video
September 4	Introduction to Semantics	
September 7	Inductive Definitions	slides notes video
September 9	Properties and Inductive Proofs	same slides same notes video

Operational Semantics

Date	Topic	Notes
September 11	Inductive Proof and Large-Step Semantics	slides notes video
September 14	The IMP Language	
September 16	IMP Properties	slides notes video
September 18	More IMP Proofs	same slides same notes whiteboard-1 whiteboard-2 video-1 video-2

Denotational Semantics

Date	Topic	Notes
September 21	Denotational Semantics	slides notes video
September 23	Denotational	
September 25	Axiomatic Semantics	slides notes video

Axiomatic Semantics

Date	Topic	Notes
September 28	Hoare Logic	slides notes video
September 30	Hoare Logic Exam	
October 2	Weakest Preconditions	slides notes video

λ-Calculus

Date	Topic	Notes
October 5	λ-Calculus	notes video
October 7	More λ-Calculus and Substitution	
October 9	de Bruijn and Combinators	
October 12	Encodings	slides notes video
October 14	**Mid-semester break**	

λ-Calculus Programming

Date	Topic	Notes	
October 16	Encodings and Fixed-Point Combinators		
October 19	Definitional Translation		HW4 due
October 21	Continuations	slides notes video	

Simple Types

Date	Topic	Notes	
October 23	Types	slides notes video	
October 26	More Types	same slides same notes video	HW5 due
October 28	Proving Type Soundness		
October 30	More Proving Type Soundness		
November 2	Normalization	slides notes video	HW6 due

Advanced Types

Date	Topic	Notes	
November 4	Advanced Types	slides notes video	
November 6	Polymorphism	slides notes video	
November 9	Type Inference		Take-Home Prelim 2
November 11	Recursive Types	slides notes video	
November 13	Records and Subtyping	slides notes video	Project Alpha due

Date	Topic
November 16	**Study days**
November 18	**Semi-finals**
November 20	**Semi-finals**
November 23	**Semi-finals**
November 25	**No class (Thanksgiving)**
November 27	**No class (Thanksgiving)**

Logic and Proof

Date	Topic	Notes	
November 30	Existential Types	slides notes video	
December 2	Propositions as Types		
December 4	Dependent Types		Project Beta due
December 7	Concurrency	slides video	Homework 7 due

Advanced Topics

Date	Topic	Notes	
December 9	Probabilistic Semantics		
December 11	After 4110		
December 14	DSLs and Bidirectional Programming	slides video	Homework 8 due
December 16	Victory Lap		Final Project due

Semester in Review

CS 4110 Home Project Resources Schedule Syllabus CMS Campuswire

Schedule

Mathematical Preliminaries

Date	Topic	Notes	
September 2	Course Overview	slides video	
September 4	Introduction to Semantics	slides notes video	
September 7	Inductive Definitions	slides notes video	
September 9	Properties and Inductive Proofs	same slides same notes video	

Operational Semantics

Date	Topic	Notes	
September 11	Inductive Proof and Large-Step Semantics	slides notes video	
September 14	The IMP Language	slides notes video	
September 16	IMP Properties	slides notes video	
September 18	More IMP Proofs	same slides same notes whiteboard-1 whiteboard-2 video-1 video-2	

Denotational Semantics

Date	Topic	Notes	
September 21	Denotational Semantics	slides notes video	
September 23	Denotational Semantics Examples	slides notes video	
September 25	Axiomatic Semantics	slides notes video	

Axiomatic Semantics

Date	Topic	Notes	
September 28	Hoare Logic	slides notes video	
September 30	Hoare Logic Examples	slides notes video	
October 2	Weakest Preconditions	slides notes video	

λ-Calculus

Date	Topic	Notes	
October 5	λ-Calculus	notes video	
October 7	More λ-Calculus and Substitution	slides notes video	
October 9	de Bruijn and Combinators	slides notes video	
October 12	Encodings	slides notes video	
October 14	**Mid-semester break**		

λ-Calculus Programming

Date	Topic	Notes	
October 16	Encodings and Fixed-Point Combinators	slides notes video	
October 19	Definitional Translation	slides notes video	HW4 due
October 21	Continuations	slides notes video	

Simple Types

Date	Topic	Notes	
October 23	Types	slides notes video	
October 26	More Types	same slides same notes video	HW5 due
October 28	Proving Type Soundness	slides notes video	
October 30	More Proving Type Soundness	same slides same notes video	
November 2	Normalization	slides notes video	HW6 due

Advanced Types

Date	Topic	Notes	
November 4	Advanced Types	slides notes video	
November 6	Polymorphism	slides notes video	
November 9	Type Inference	slides notes video	Take-Home Prelim 2
November 11	Recursive Types	slides notes video	
November 13	Records and Subtyping	slides notes video	Project Alpha due
November 16	**Study days**		
November 18	**Semi-finals**		
November 20	**Semi-finals**		
November 23	**Semi-finals**		
November 25	**No class (Thanksgiving)**		
November 27	**No class (Thanksgiving)**		

Logic and Proof

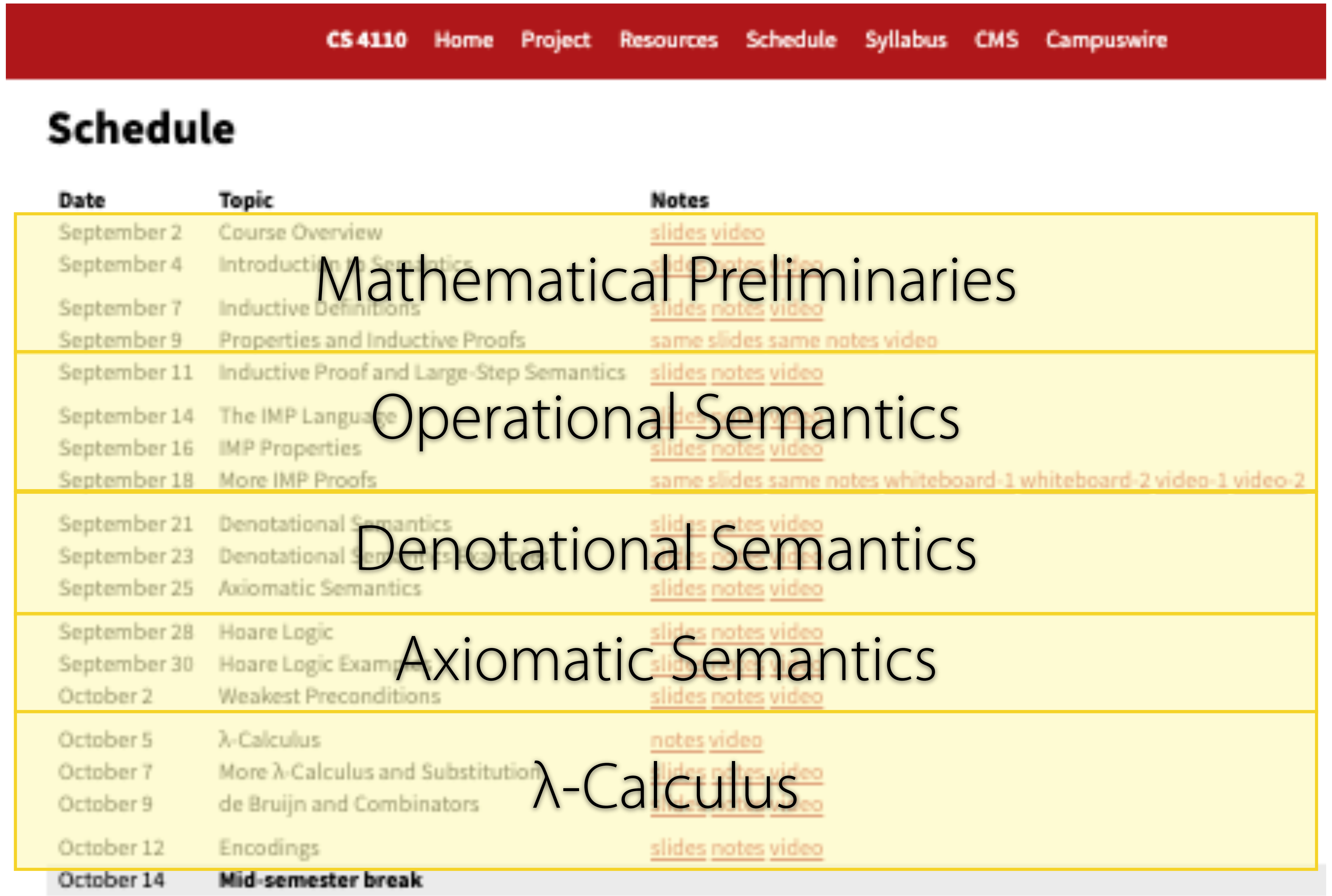
Date	Topic	Notes	
November 30	Existential Types	slides notes video	
December 2	Propositions as Types	slides notes video	
December 4	Dependent Types	slides notes video	Project Beta due
December 7	Concurrency	slides video	Homework 7 due

Advanced Topics

Date	Topic	Notes	
December 9	Probabilistic Semantics		
December 11	After 4110		
December 14	DSLs and Bidirectional Programming	slides video	Homework 8 due
December 16	Victory Lap		Final Project due

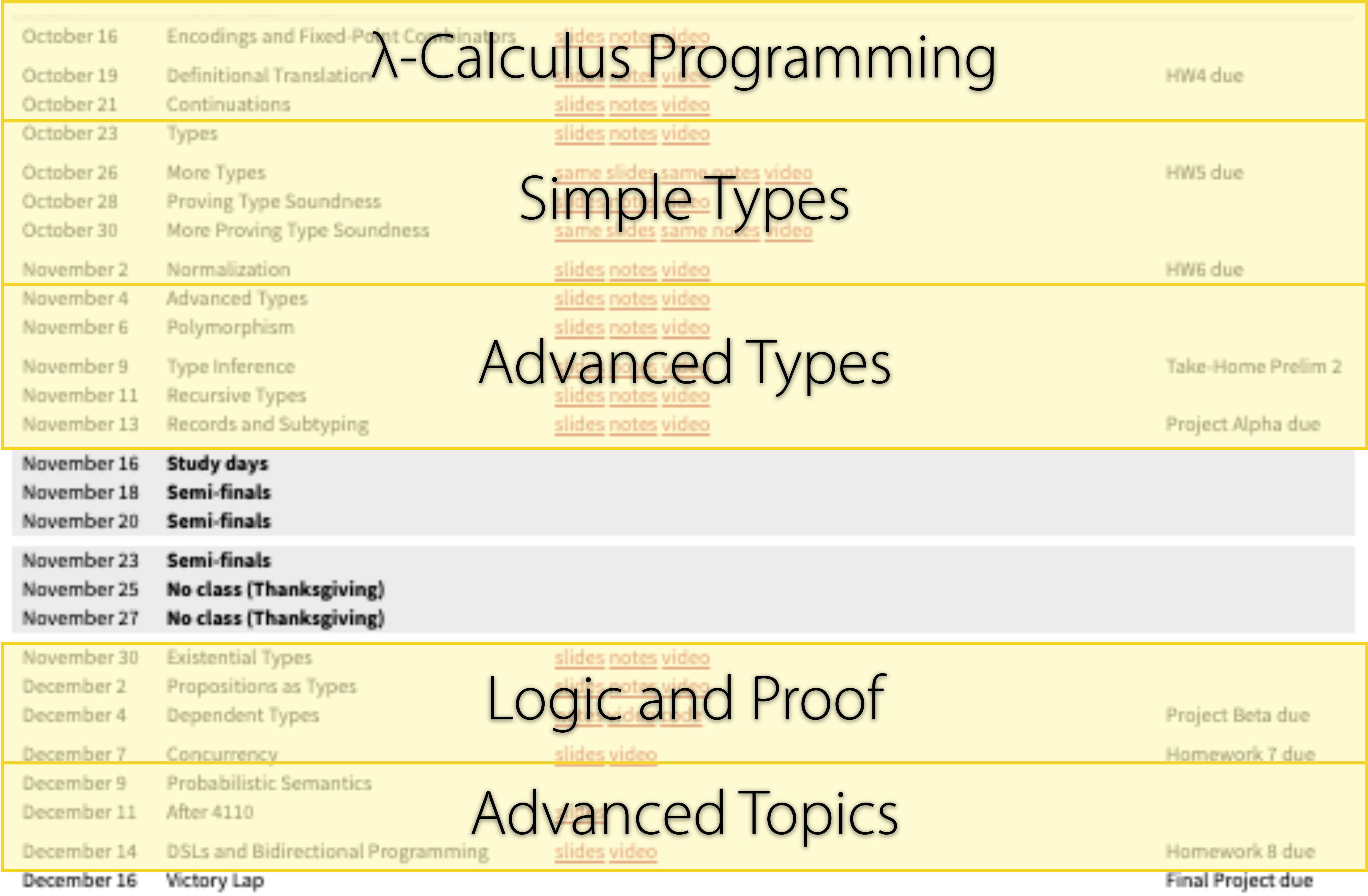
Proofs

Semester in Review


CS 4110 Home Project Resources Schedule Syllabus CMS Campuswire

Schedule

Date	Topic	Notes
September 2	Course Overview	slides video
September 4	Introduction to Semantics	
September 7	Inductive Definitions	slides notes video
September 9	Properties and Inductive Proofs	same slides same notes video
September 11	Inductive Proof and Large-Step Semantics	slides notes video
September 14	The IMP Language	
September 16	IMP Properties	slides notes video
September 18	More IMP Proofs	same slides same notes whiteboard-1 whiteboard-2 video-1 video-2
September 21	Denotational Semantics	slides notes video
September 23	Denotational Semantics	
September 25	Axiomatic Semantics	slides notes video
September 28	Hoare Logic	slides notes video
September 30	Hoare Logic Examples	
October 2	Weakest Preconditions	slides notes video
October 5	λ-Calculus	notes video
October 7	More λ-Calculus and Substitution	
October 9	de Bruijn and Combinators	
October 12	Encodings	slides notes video
October 14	**Mid-semester break**	



Date	Topic	Notes	
October 16	Encodings and Fixed-Point Combinators	slides notes video	
October 19	Definitional Translation		HW4 due
October 21	Continuations	slides notes video	
October 23	Types	slides notes video	
October 26	More Types	same slides same notes video	HW5 due
October 28	Proving Type Soundness		
October 30	More Proving Type Soundness	same slides same notes video	
November 2	Normalization	slides notes video	HW6 due
November 4	Advanced Types	slides notes video	
November 6	Polymorphism	slides notes video	
November 9	Type Inference		Take-Home Prelim 2
November 11	Recursive Types	slides notes video	
November 13	Records and Subtyping	slides notes video	Project Alpha due
November 16	**Study days**		
November 18	**Semi-finals**		
November 20	**Semi-finals**		
November 23	**Semi-finals**		
November 25	**No class (Thanksgiving)**		
November 27	**No class (Thanksgiving)**		
November 30	Existential Types	slides notes video	
December 2	Propositions as Types		
December 4	Dependent Types		Project Beta due
December 7	Concurrency	slides video	Homework 7 due
December 9	Probabilistic Semantics		
December 11	After 4110		
December 14	DSLs and Bidirectional Programming	slides video	Homework 8 due
December 16	Victory Lap		Final Project due

Proofs

Interpreters

Semester in Review

CS 4110 Home Project Resources Schedule Syllabus CMS Campuswire

Schedule

Mathematical Preliminaries

Date	Topic	Notes
September 2	Course Overview	slides video
September 4	Introduction to Semantics	slides notes video
September 7	Inductive Definitions	slides notes video
September 9	Properties and Inductive Proofs	same slides same notes video

Operational Semantics

Date	Topic	Notes
September 11	Inductive Proof and Large-Step Semantics	slides notes video
September 14	The IMP Language	slides notes video
September 16	IMP Properties	slides notes video
September 18	More IMP Proofs	same slides same notes whiteboard-1 whiteboard-2 video-1 video-2

Denotational Semantics

Date	Topic	Notes
September 21	Denotational Semantics	slides notes video
September 23	Denotational Semantics	slides notes video
September 25	Axiomatic Semantics	slides notes video

Axiomatic Semantics

Date	Topic	Notes
September 28	Hoare Logic	slides notes video
September 30	Hoare Logic Examples	slides notes video
October 2	Weakest Preconditions	slides notes video

λ-Calculus

Date	Topic	Notes
October 5	λ-Calculus	notes video
October 7	More λ-Calculus and Substitution	slides notes video
October 9	de Bruijn and Combinators	slides notes video
October 12	Encodings	slides notes video
October 14	**Mid-semester break**	

λ-Calculus Programming

Date	Topic	Notes	
October 16	Encodings and Fixed-Point Combinators	slides notes video	
October 19	Definitional Translation	slides notes video	HW4 due
October 21	Continuations	slides notes video	

Simple Types

Date	Topic	Notes	
October 23	Types	slides notes video	
October 26	More Types	same slides same notes video	HW5 due
October 28	Proving Type Soundness	slides notes video	
October 30	More Proving Type Soundness	same slides same notes video	
November 2	Normalization	slides notes video	HW6 due

Advanced Types

Date	Topic	Notes	
November 4	Advanced Types	slides notes video	
November 6	Polymorphism	slides notes video	
November 9	Type Inference	slides notes video	Take-Home Prelim 2
November 11	Recursive Types	slides notes video	
November 13	Records and Subtyping	slides notes video	Project Alpha due
November 16	**Study days**		
November 18	**Semi-finals**		
November 20	**Semi-finals**		
November 23	**Semi-finals**		
November 25	**No class (Thanksgiving)**		
November 27	**No class (Thanksgiving)**		

Logic and Proof

Date	Topic	Notes	
November 30	Existential Types	slides notes video	
December 2	Propositions as Types	slides notes video	
December 4	Dependent Types	slides notes video	Project Beta due
December 7	Concurrency	slides video	Homework 7 due

Advanced Topics

Date	Topic	Notes	
December 9	Probabilistic Semantics		
December 11	After 4110		
December 14	DSLs and Bidirectional Programming	slides video	Homework 8 due
December 16	Victory Lap		Final Project due

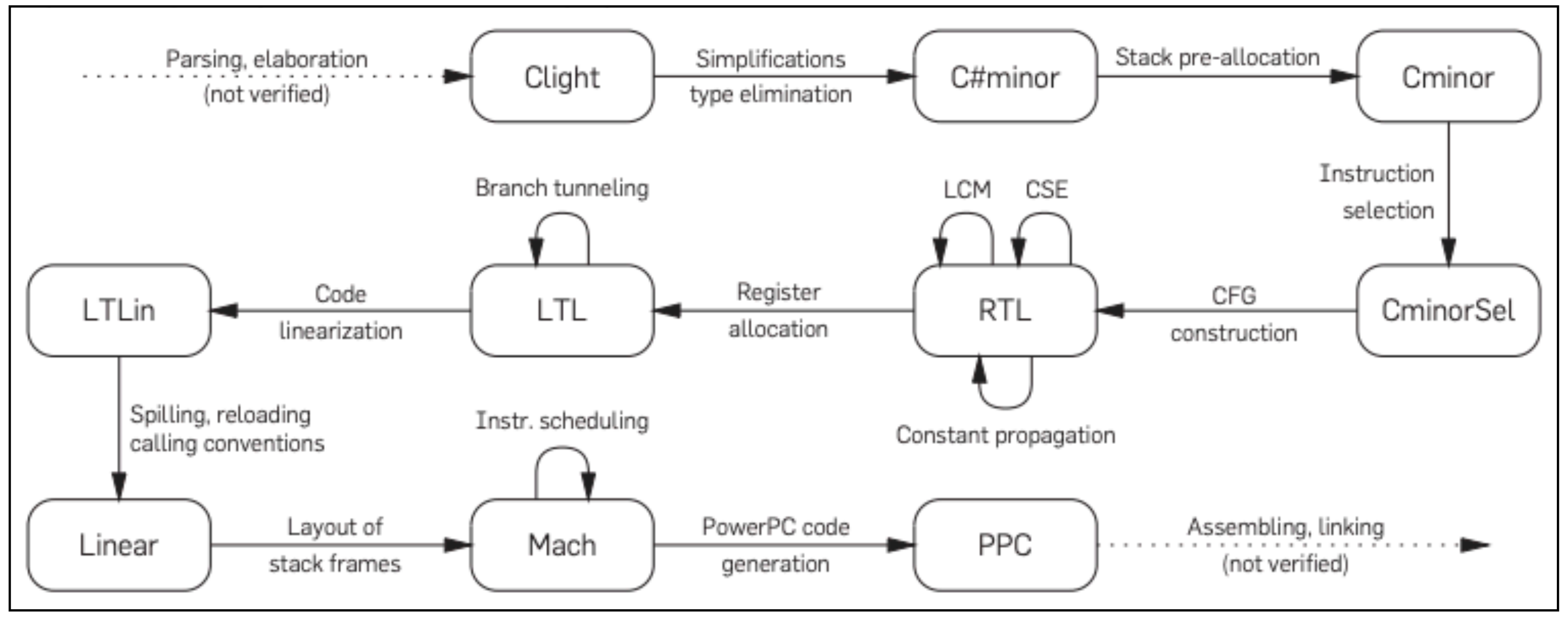
Proofs

Interpreters

Verifiers

Trend: Verified Software

- Researchers are increasingly developing operational models of real-world languages
- Progress in verification makes it possible to build end-to-end verified systems
- Example: CompCert
  - Formal semantics for (a large subset of C)
  - Formal semantics for PPC (and later x86)
  - Full compiler verified in Coq



DOI:10.1145/1538788.1538814

Formal Verification of a Realistic Compiler

By Xavier Leroy

Abstract

This paper reports on the development and formal verification (proof of semantic preservation) of CompCert, a compiler from Clight (a large subset of the C programming language) to PowerPC assembly code, using the Coq proof assistant both for programming the compiler and for proving its correctness. Such a verified compiler is useful in the context of critical software and its formal verification: the verification of the compiler guarantees that the safety properties proved on the source code hold for the executable compiled code as well.

1. INTRODUCTION

Can you trust your compiler? Compilers are generally assumed to be semantically transparent: the compiled code should behave as prescribed by the semantics of the source program. Yet, compilers—and especially optimizing compilers—are complex programs that perform complicated symbolic transformations. Despite intensive testing, bugs in compilers do occur, causing the compilers to crash at compile-time or—much worse—to silently generate an incorrect executable for a correct source program.

For low-assurance software, validated only by testing, the impact of compiler bugs is low: what is tested is the executable code produced by the compiler; rigorous testing should expose compiler-introduced errors along with errors already present in the source program. Note, however, that compiler-introduced bugs are notoriously difficult to expose and track down. The picture changes dramatically for safety-critical, high-assurance software. Here, validation by testing reaches its limits and needs to be complemented or

preserves the semantics of the source programs. For the last 5 years, we have been working on the development of a *realistic*, *verified* compiler called CompCert. By *verified*, we mean a compiler that is accompanied by a machine-checked proof of a semantic preservation property: the generated machine code behaves as prescribed by the semantics of the source program. By *realistic*, we mean a compiler that could realistically be used in the context of production of critical software. Namely, it compiles a language commonly used for critical embedded software: neither Java nor ML nor assembly code, but a large subset of the C language. It produces code for a processor commonly used in embedded systems: we chose the PowerPC because it is popular in avionics. Finally, the compiler must generate code that is efficient enough and compact enough to fit the requirements of critical embedded systems. This implies a multipass compiler that features good register allocation and some basic optimizations.

Proving the correctness of a compiler is by no ways a new idea: the first such proof was published in 1967[16] (for the compilation of arithmetic expressions down to stack machine code) and mechanically verified in 1972.[17] Since then, many other proofs have been conducted, ranging from single-pass compilers for toy languages to sophisticated code optimizations.[8] In the CompCert experiment, we carry this line of work all the way to end-to-end verification of a complete compilation chain from a structured imperative language down to assembly code through eight intermediate languages. While conducting the verification of CompCert, we found that many of the nonoptimizing translations performed, while often considered obvious in the compiler literature, are surprisingly tricky to formally prove correct.

This paper gives a high-level overview of the CompCert

To appear at POPL '21

Petr4: Formal Foundations for P4 Data Planes

RYAN DOENGES, Cornell University, USA
MINA TAHMASBI ARASHLOO, Cornell University, USA
ALEXANDER CHANG, Cornell University, USA
NEWTON NI, Cornell University, USA
SAMWISE PARKINSON, Cornell University, USA
RUDY PETERSON, Cornell University, USA
ALAIA SOLKO-BRESLIN, Cornell University, USA
AMANDA XU, Cornell University, USA
NATE FOSTER, Cornell University, USA

P4 is a domain-specific language for specifying the behavior of packet-processing systems. It is based on an elegant design with high-level abstractions, such as parsers and match-action pipelines, which can be compiled to efficient implementations in hardware or software. Unfortunately, like many industrial languages, P4 lacks a formal foundation. The P4 specification is a 160-page document with a mixture of informal prose, graphical diagrams, and pseudocode. The reference compiler is complex, running to over 40KLoC of C++ code. Clearly neither of these artifacts is suitable for formal reasoning.

This paper presents a new framework, called Petr4, that puts P4 on a solid foundation. Petr4 uses standard elements of the semantics engineering toolkit, namely type systems and operational semantics, to build a compositional semantics that assigns an unambiguous meaning to every P4 program. Petr4 is implemented as an OCaml prototype that has been validated against a suite of over 750 tests from the reference implementation. While developing Petr4, we discovered dozens of bugs in the language specification and the reference implementation, many of which have been fixed. Furthermore, we have used Petr4 to establish the soundness of P4's type system, prove key properties such as termination, and formalize a language extension.

The Petr4 Team

Ryan Doenges

Mina Tahmasbi Arashloo

Santiago Bautista

Alexander Chang

Newton Ni

Samwise Parkinson

Rudy Peterson

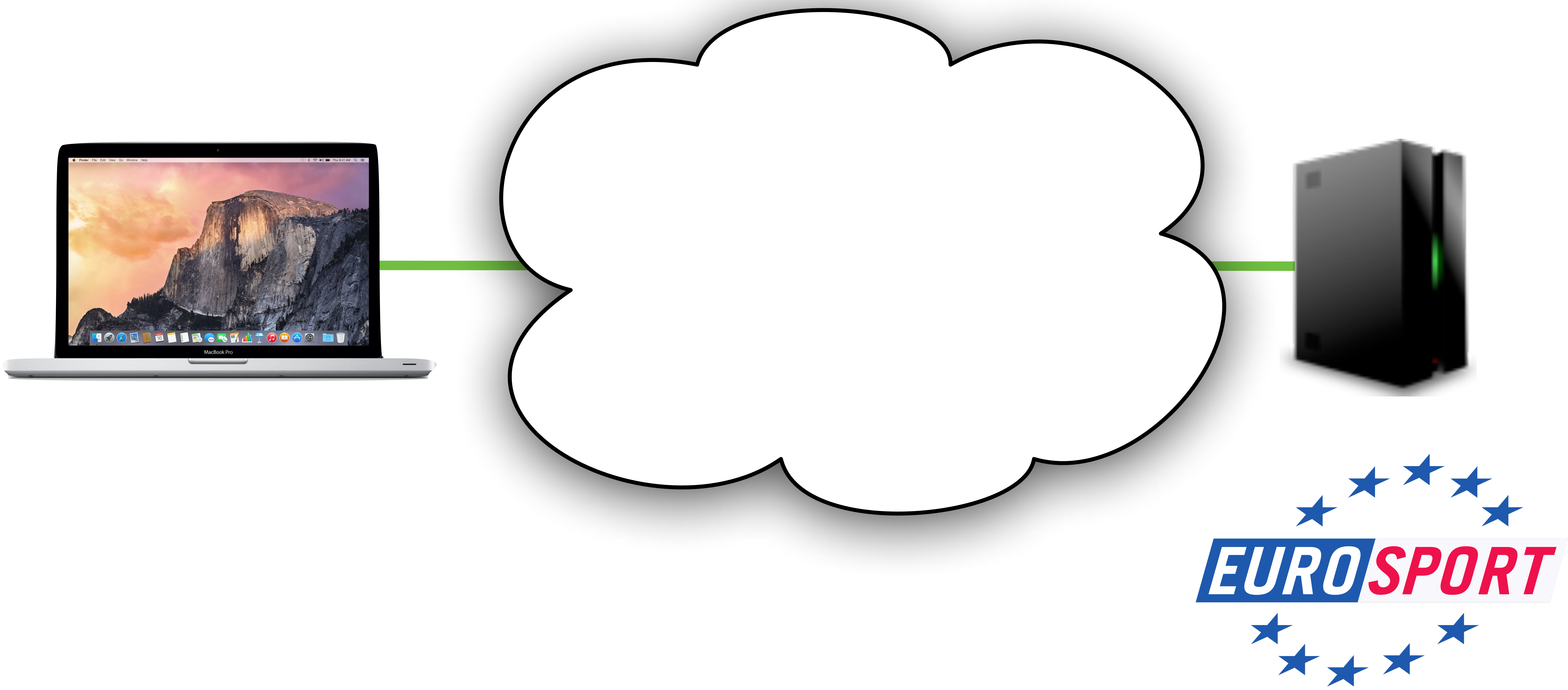
Alaia Solko-Breslin

Amanda Xu

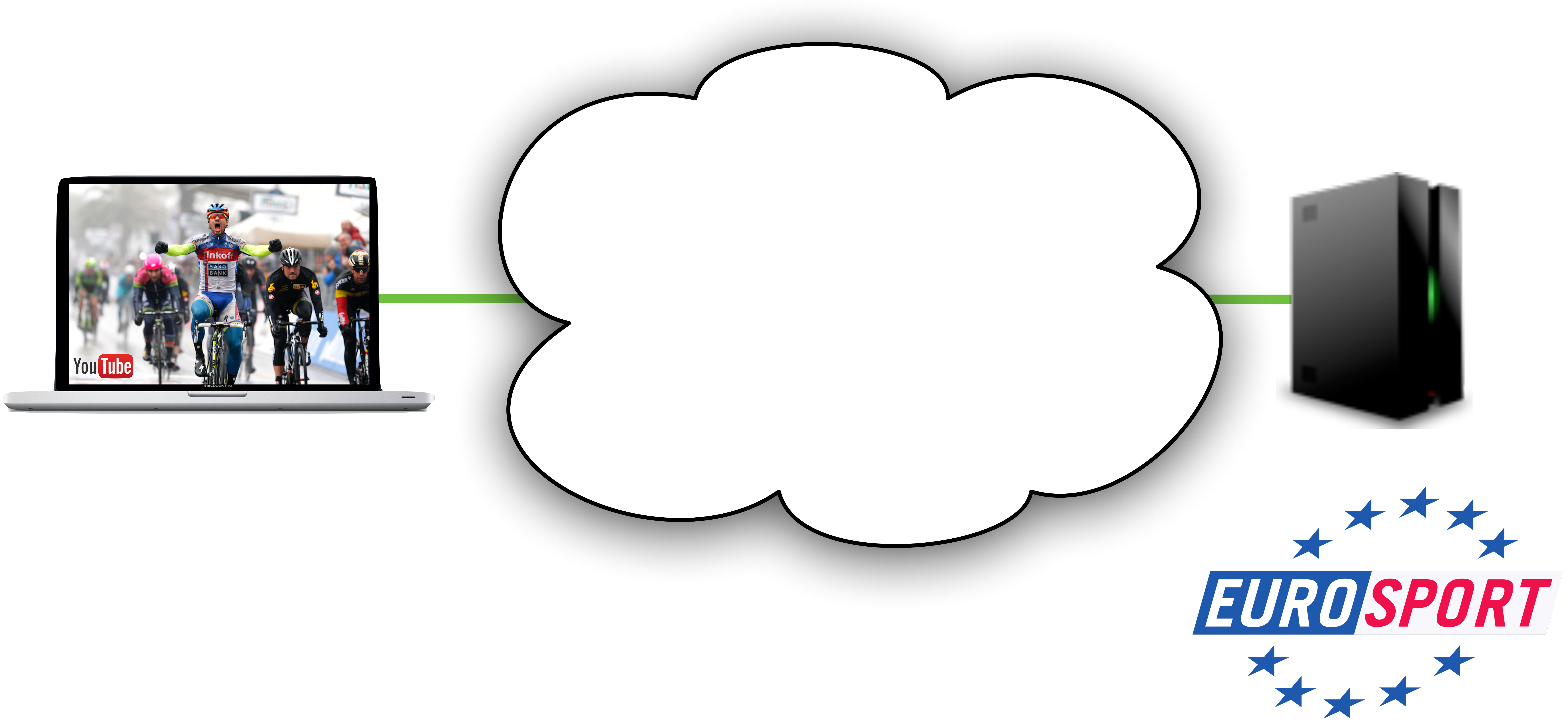
Nate Foster

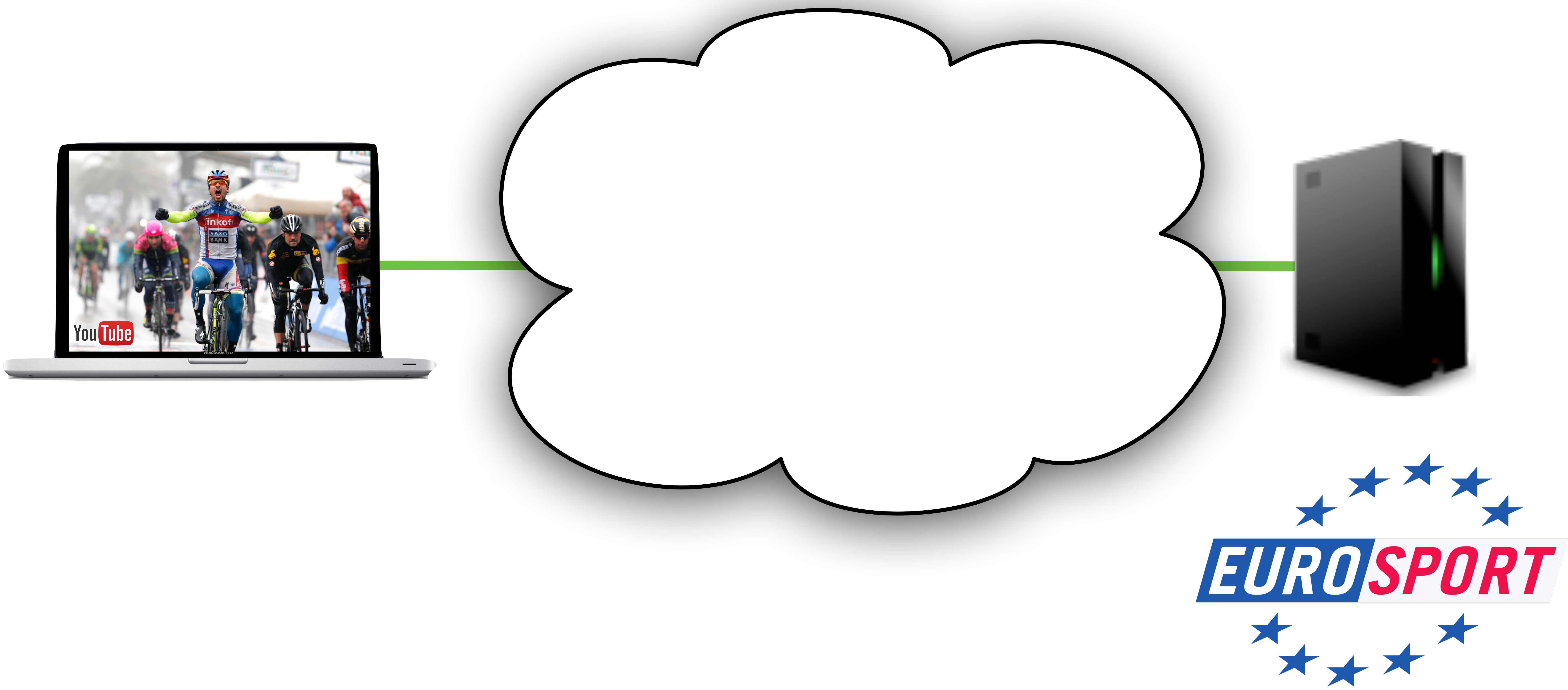
Let's see what's going on in the Tour this morning!

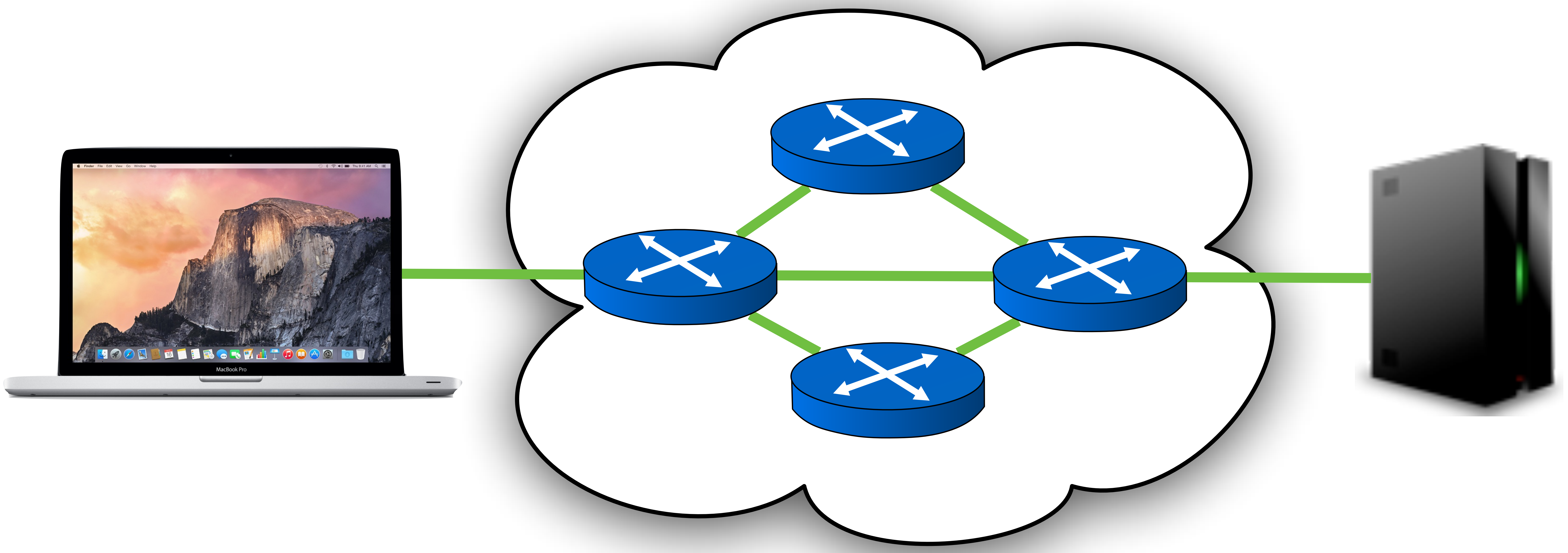

MacBook Pro


EUROSPORT

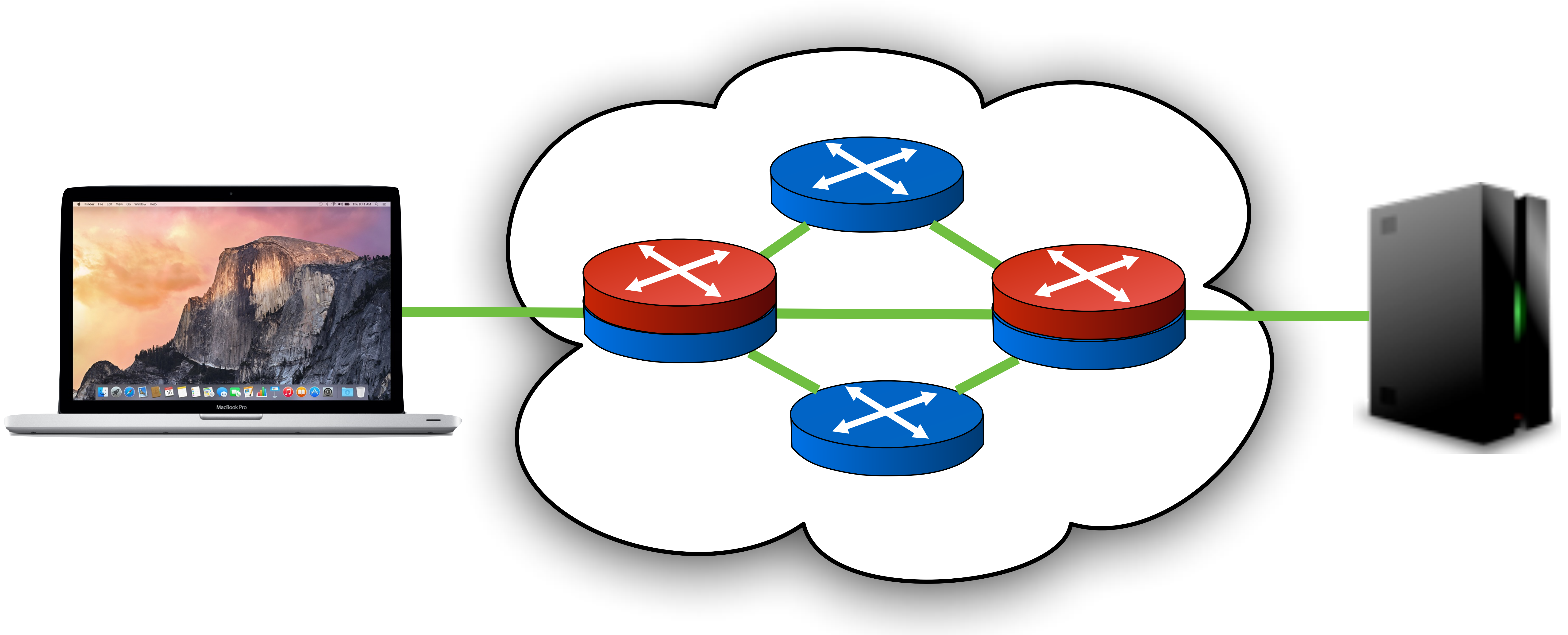

YouTube
EUROSPORT


YouTube
EUROSPORT

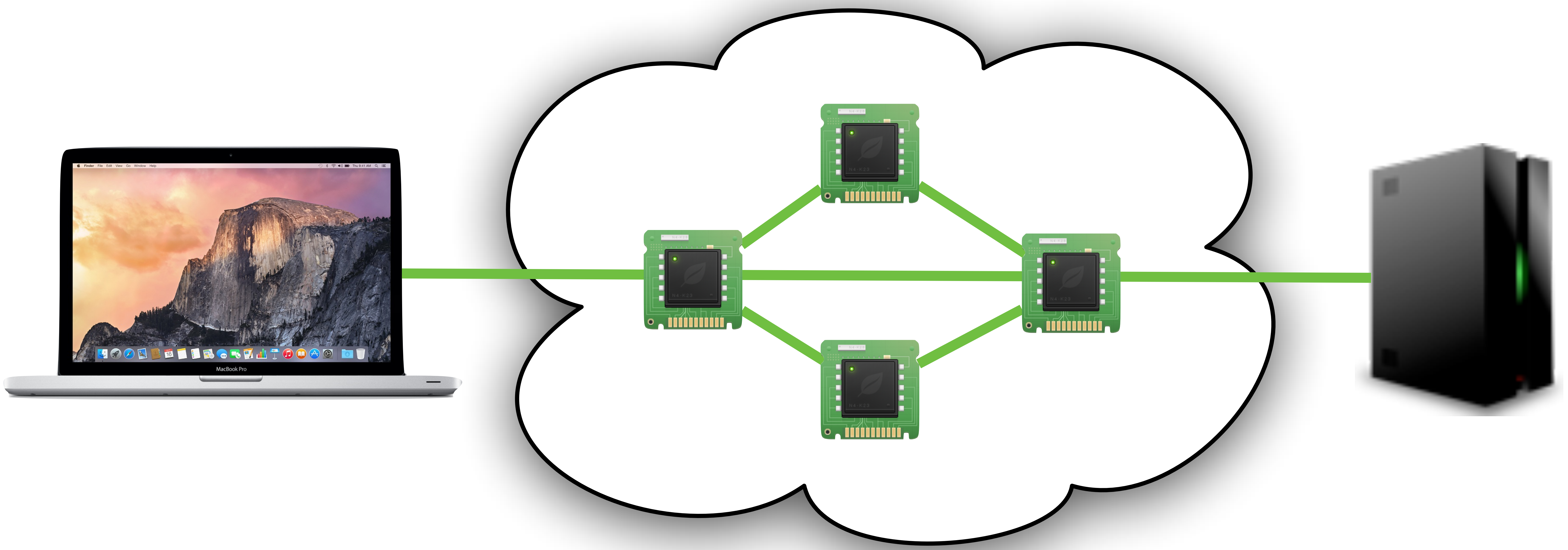

YouTube
EUROSPORT



Data plane: forward packets, balances load, implements monitoring, etc.

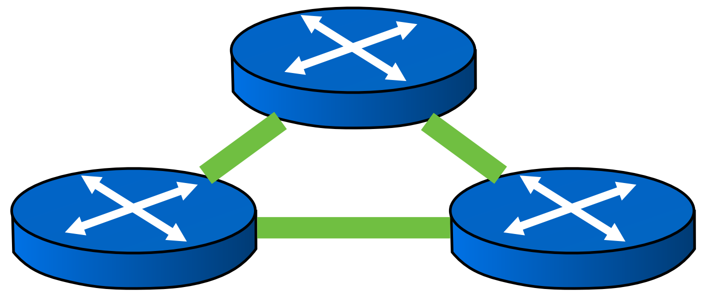


Control Plane: discovers topology, computes routes, enforces policies, etc.

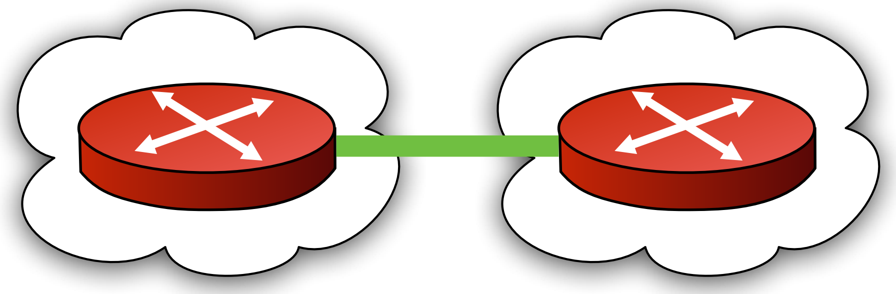


Network Devices: implement packet processing, buffering, queueing, etc. at line rate

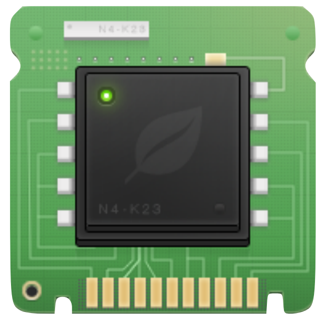
Network Programming Challenges



Networks are *distributed* systems with thousands of interacting nodes



Networks enforce complex *security* policies that span trust boundaries



Networks are expected to offer good *performance* with limited resources

Brief History of Network Verification

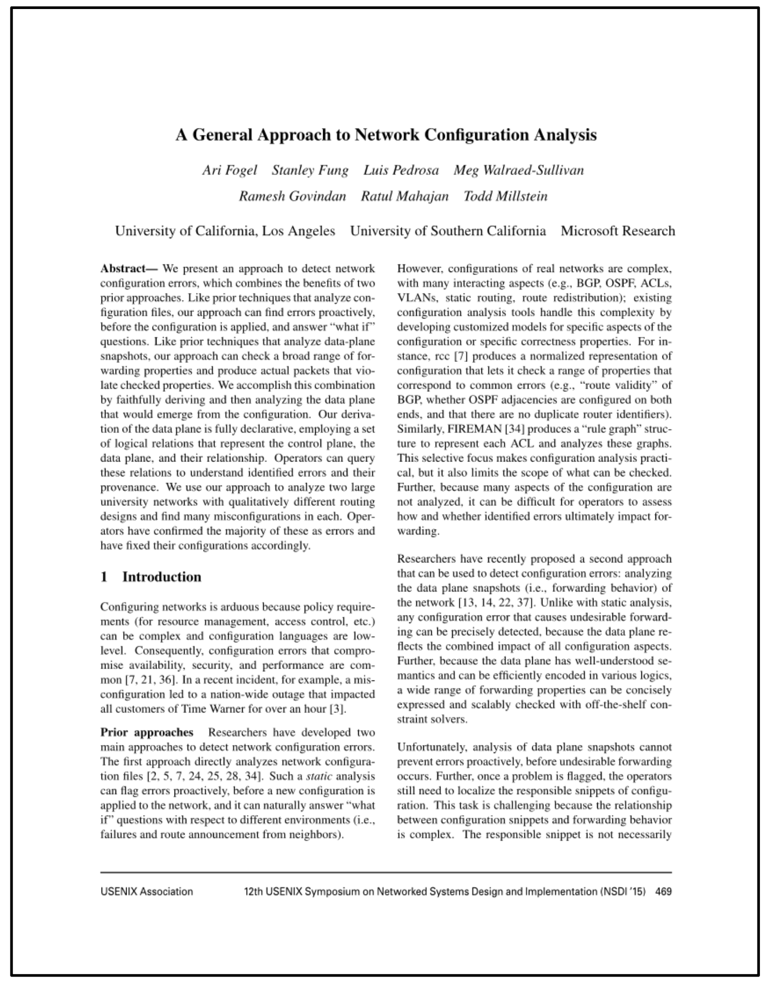
On Static Reachability Analysis of IP Networks

Geoffrey G. Xie[*] Jibin Zhan[†] David A. Maltz[†] Hui Zhang[†] Albert Greenberg[‡] Gisli Hjalmtysson[‡] Jennifer Rexford[‡]

Header Space Analysis: Static Checking For Networks

Peyman Kazemian, Stanford University · George Varghese, UCSD and Yahoo! Research · Nick McKeown, Stanford University

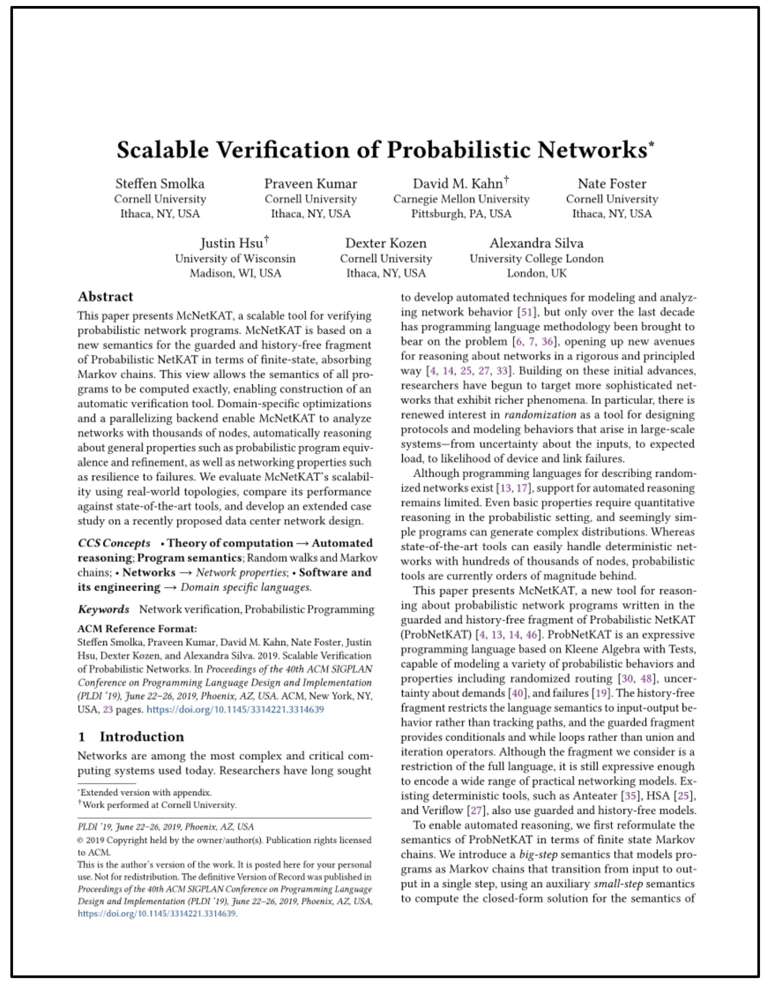
...twork-Wide Invariants in Real Time



A General Approach to Network Configuration Analysis

Ari Fogel, Stanley Fung, Luis Pedrosa, Meg Walraed-Sullivan, Ramesh Govindan, Ratul Mahajan, Todd Millstein

University of California, Los Angeles · University of Southern California · Microsoft Research



Scalable Verification of Probabilistic Networks[*]

Steffen Smolka, Praveen Kumar, David M. Kahn[†], Nate Foster, Justin Hsu[†], Dexter Kozen, Alexandra Silva

Static reachability for IP networks

Software-defined networks

Distributed control planes

Probabilistic networks

1996

2010

2020

Status Quo

Network Working Group
Request for Comments: 2418
Obsoletes: 1603
BCP: 25
Category: Best Current Practice

S. Bradner
Editor
Harvard University
September 1998

IETF Working Group
Guidelines and Procedures

“We believe in rough consensus and running code”

Abstract

The Internet Engineering Task Force (IETF) has responsibility for developing and reviewing specifications intended as Internet Standards. IETF activities are organized into working groups (WGs). This document describes the guidelines and procedures for formation and operation of IETF working groups. It also describes the formal relationship between IETF participants WG and the Internet Engineering Steering Group (IESG) and the basic duties of IETF participants, including WG Chairs, WG participants, and IETF Area Directors.

Shaky Foundations

Bill Joy

"The system doesn't have a semantics; in a very deep sense the program does not have a meaning; what it has is… it's just whatever happens. That's kind of the way nature works too, right?"

Language-Based Approach

Anatomy of a P4 Program

Types

Parser

Controls

```
// Programmer-defined types
header hop {
  bit<7> port;
  bit<1> bos;
}
struct headers {
  hop[9] hops;
}
// Programmer-defined components
parser MyParse(packet_in pkt,
               out headers hdrs,
               inout std_meta meta) {
  state start {
    pkt.extract(hdrs.hops.next);
    transition select(hdrs.hops.last.bos) {
      1: accept;
      default: start;
    }
  }
}
control MyPipe(inout headers hdrs,
               inout std_meta meta) {
  action allow() { }
  action deny() { meta.egress_port = 0xFF; }
  table acl {
    key = { meta.ingress_port : exact;
            meta.egress_port : exact; }
    actions = { allow; deny; }
    default_action = deny();
  }
  apply {
    meta.egress_port =
      (bit<8>)hdrs.hops[0].port;
    if(!hdrs.hops[0].isValid()) exit;
    hdrs.hops.pop_front(1);
    acl.apply();
  }
}
control MyDeparse(packet_out pkt,
                  in headers hdrs) {
  apply { pkt.emit(hdrs.hops); }
}
Switch(MyParse(),MyPipe(),MyDeparse()) main;
```

Formalization Challenges

The "official" definition of P4 resides in an informal specification and a 40KLoC C++ implementation

$P4_{16}$ Language Specification

version 1.2.1

The P4 Language Consortium

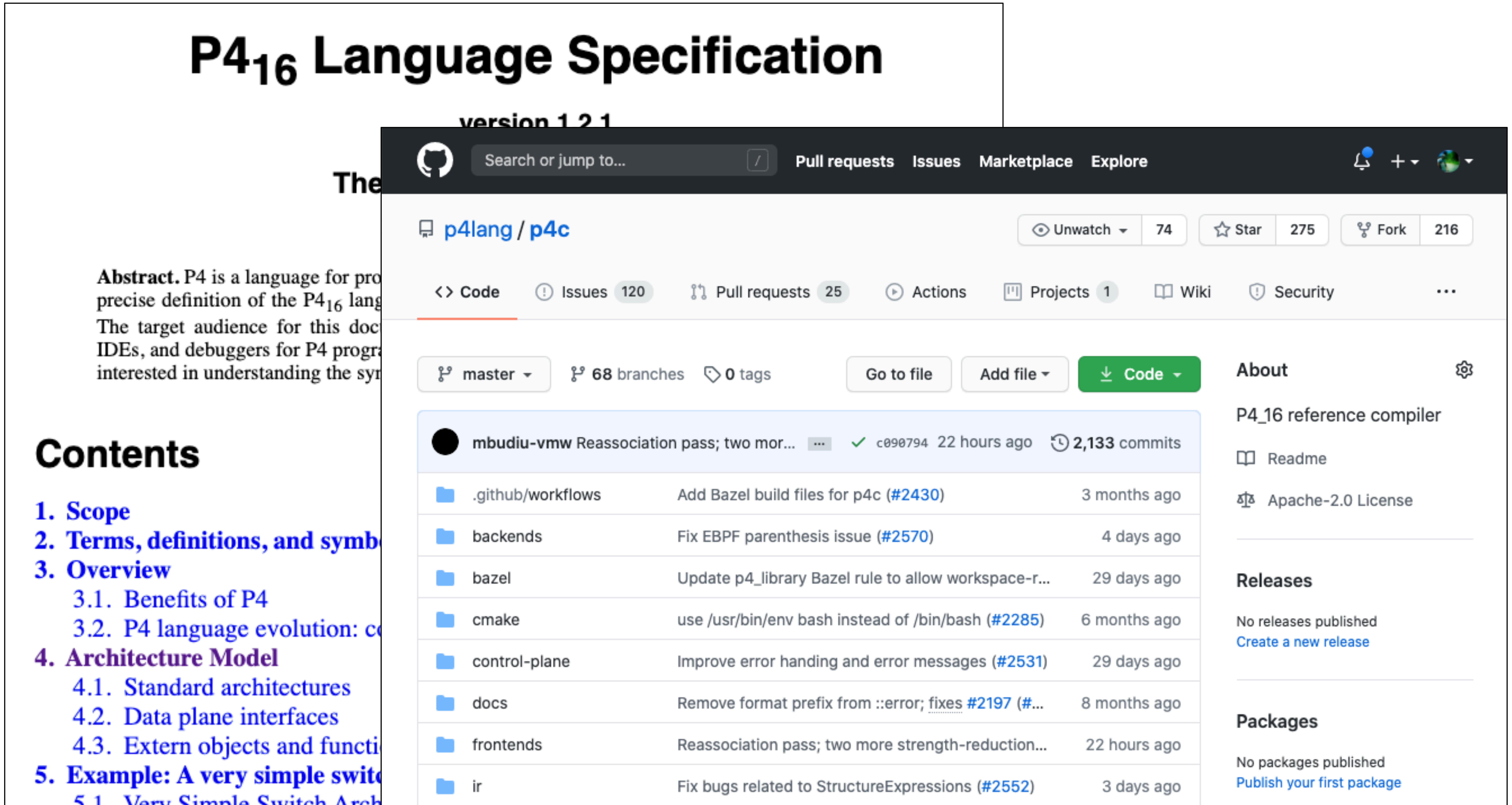
2020-06-11

Abstract. P4 is a language for programming the data plane of network devices. This document provides a precise definition of the $P4_{16}$ language, which is the 2016 revision of the P4 language (`http://p4.org`). The target audience for this document includes developers who want to write compilers, simulators, IDEs, and debuggers for P4 programs. This document may also be of interest to P4 programmers who are interested in understanding the syntax and semantics of the language at a deeper level.

Contents

Formalization Challenges

The "official" definition of P4 resides in an informal specification and a 40KLoC C++ implementation



Challenge: Undefined Values

8.22. Reading uninitialized values and writing fields of invalid headers

As mentioned in Section 8.17, any reference to an element of a header stack `hs[index]` where `index` is a compile-time constant expression must give an error at compile time if the value of the index is out of range. That section also defines the run time behavior of the expressions `hs.next` and `hs.last`, and the behaviors specified there take precedence over anything in this section for those expressions.

All mentions of header stack elements in this section only apply for expressions `hs[index]` where `index` is a run time variable expression, i.e. not a compile-time constant value. A P4 implementation is allowed not to support `hs[index]` where `index` is a run time variable expression, but if it does support these expressions, the implementation should conform to the behaviors specified in this section.

The result of reading a value in any of the situations below is that some unspecified value will be used for that field.

- reading a field from a header that is currently invalid.
- reading a field from a header that is currently valid, but the field has not been initialized since the header was last made valid.
- reading any other value that has not been initialized, e.g. a field from a `struct`, any uninitialized variable inside of an `action` or `control`, or an `out` parameter of a `control` or `action` you have called, which was not assigned a value during the execution of that `control` or `action` (this list of examples is not intended to be exhaustive).
- reading a field of a header that is an element of a header stack, where the index is out of range for the header stack.

p4c: Unsound Optimization

fruffy commented on Apr 17 • edited

Contributor

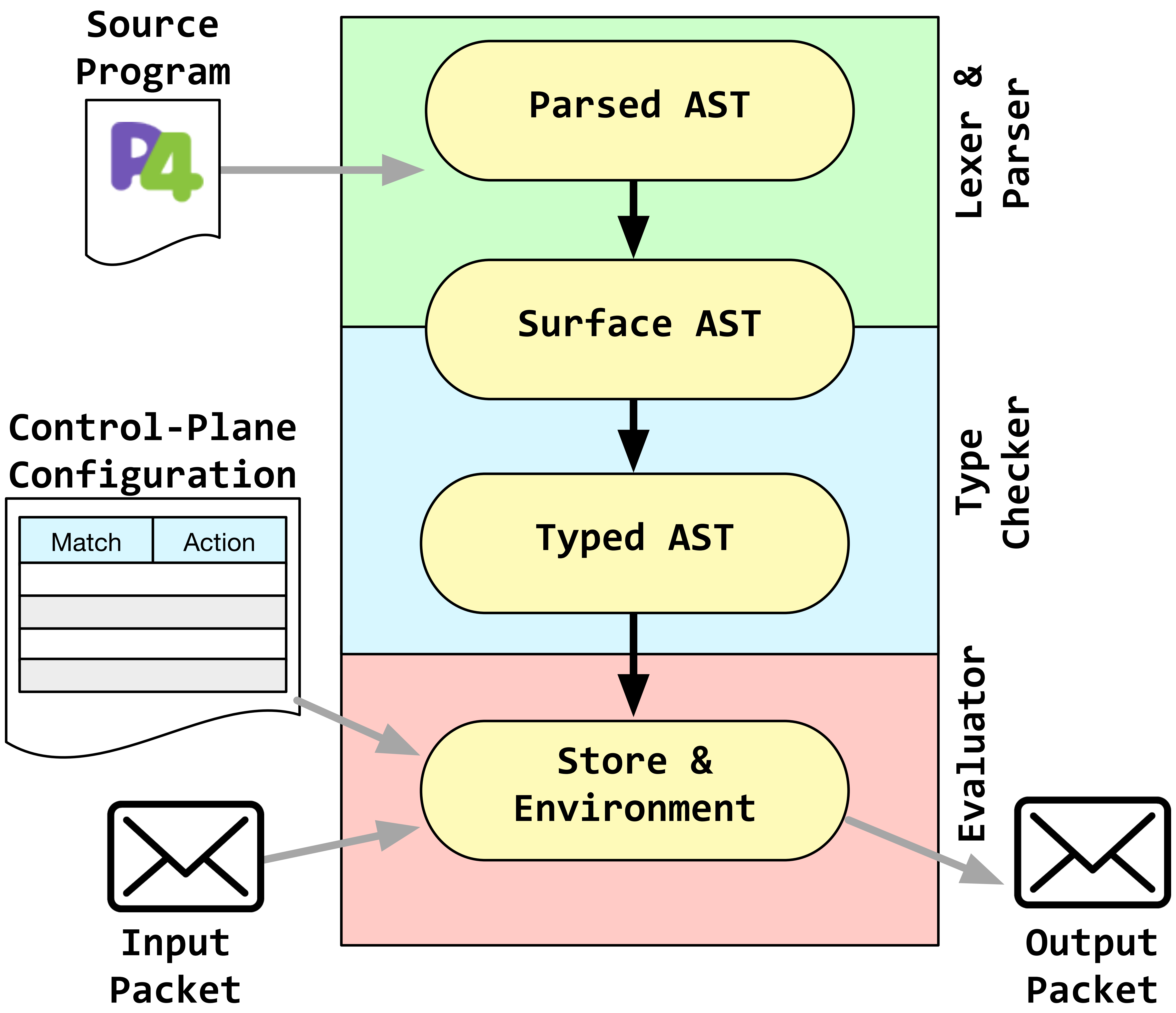
Hello.
I have another clarification question on `setInvalid`, this is a follow-up to #2212.
This issue is quite esoteric but has given me some trouble recently.

```
control ingress(inout Headers h, inout Meta m, inout standard_metadata_t sm) {
    apply {
        h.h.setInvalid();
        h.h.a = 1;
        h.eth_hdr.src_addr = h.h.a;
        if (h.eth_hdr.src_addr != 1) {
            h.h.setValid();
            h.h.a = 1;
        }
    }
}
```

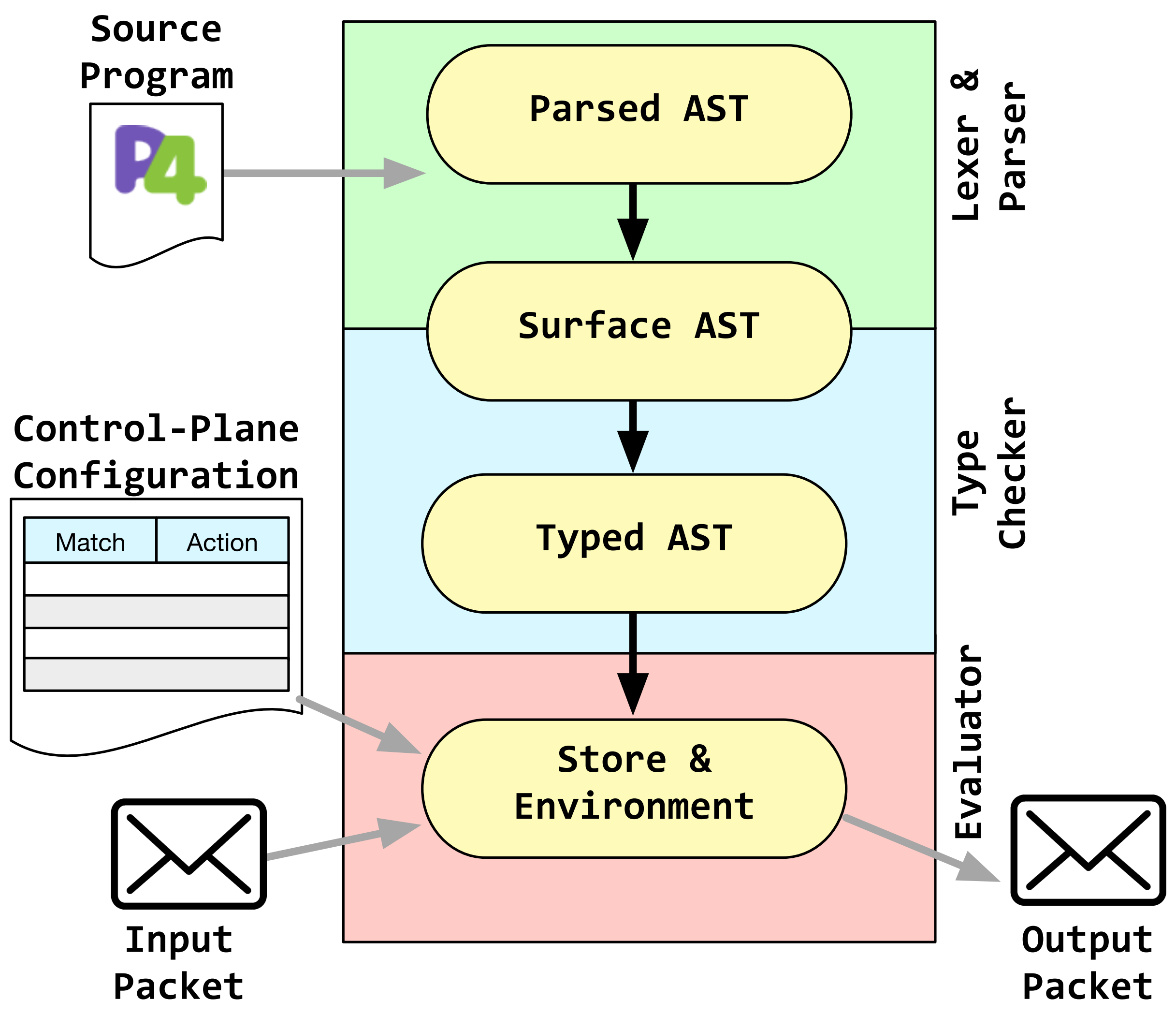
which is eventually turned into

```
control ingress(inout Headers h, inout Meta m, inout standard_metadata_t sm) {
    apply {
        h.h.setInvalid();
        h.h.a = 48w1;
        h.eth_hdr.src_addr = 48w1;
    }
}
```

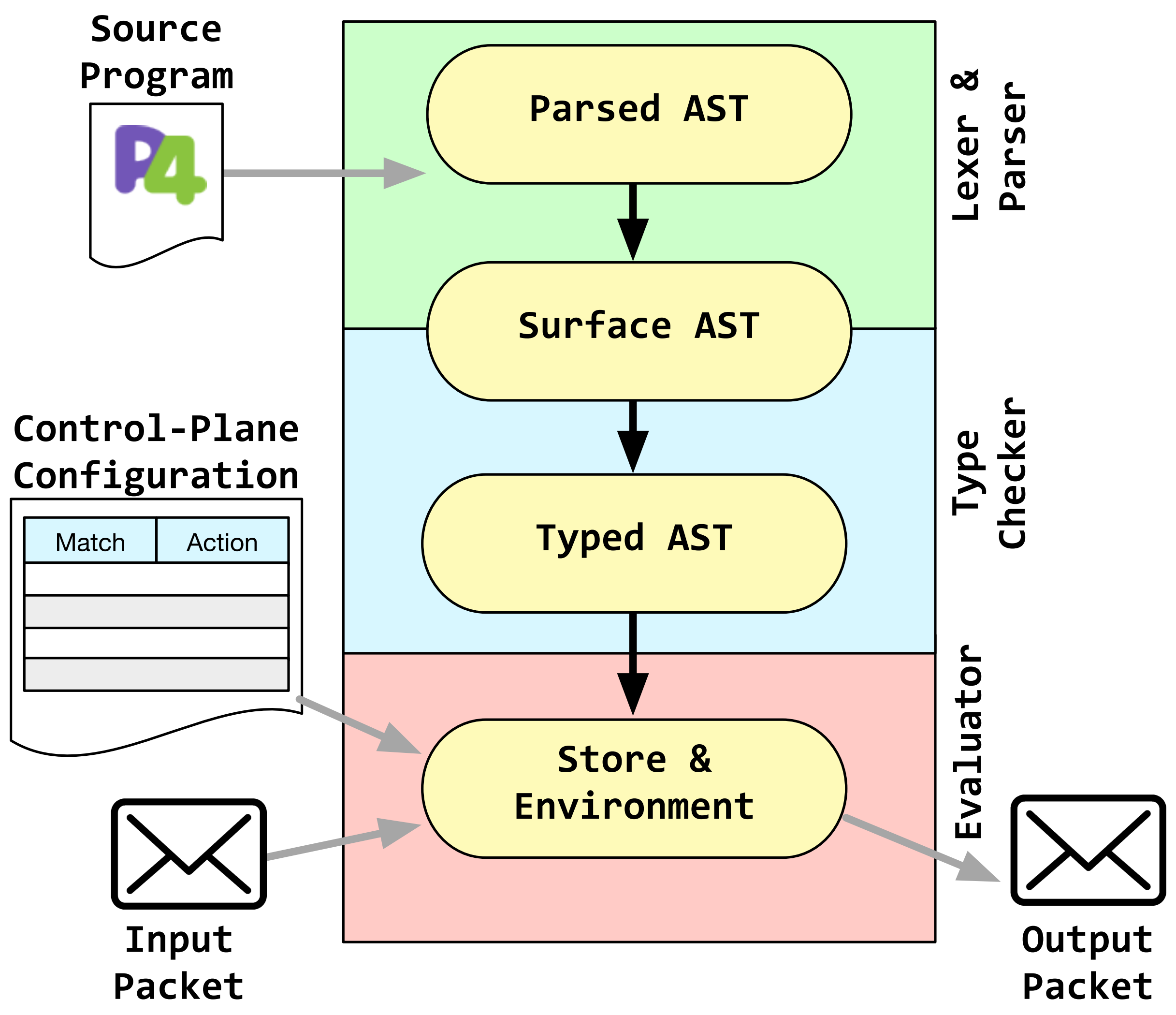
Petr4 Architecture

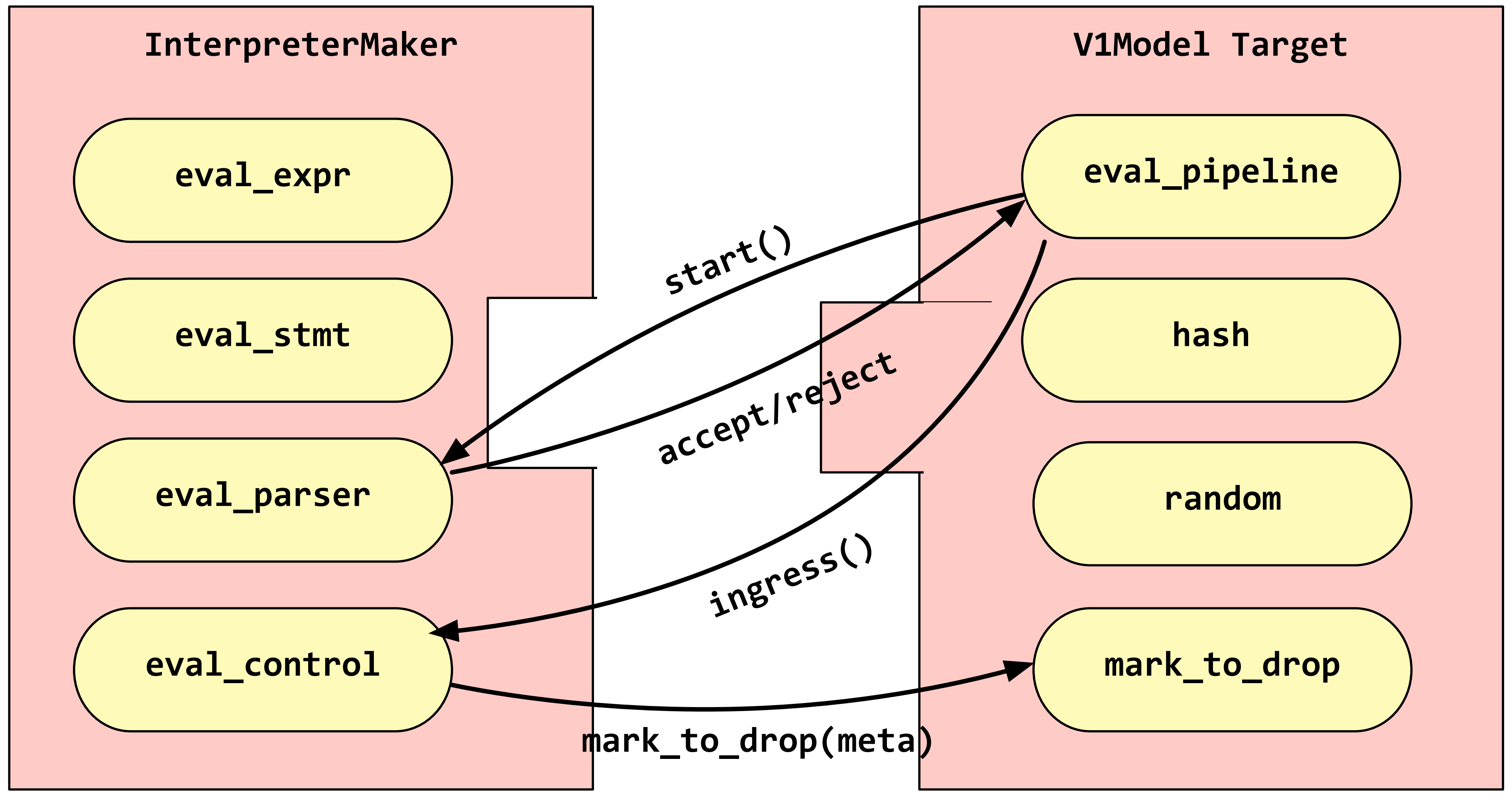


Petr4 Architecture



Petr4 Architecture





Modular design allows customizing semantics for each architecture

Petr4 Syntax

Types

$$
\begin{array}{rcll}
\rho & ::= & \mathsf{bool} & \text{booleans} \\
& | & \mathsf{int} & \text{integers} \\
& | & \mathsf{bit}\langle exp \rangle & \text{bitstrings} \\
& | & \mathsf{error}\ \{\overline{f}\} & \text{errors} \\
& | & \mathsf{match\_kind}\ \{\overline{f}\} & \text{match kinds} \\
& | & \mathsf{enum}\ X\ \{\overline{f}\} & \text{enums} \\
& | & \{\overline{f : \tau}\} & \text{records} \\
& | & \mathsf{header}\ \{\overline{f : \tau}\} & \text{headers} \\
& | & \tau[n] & \text{stacks} \\
& | & X & \text{type variables}
\end{array}
$$

$$
\begin{array}{rcll}
\tau & ::= & \rho & \text{data types} \\
& | & \mathsf{table} & \text{tables} \\
& | & \mathsf{function}\langle \overline{X} \rangle (\overline{d\ x : \rho}) \to \rho_{ret} & \text{functions} \\
& | & \mathsf{ctor}(\overline{x : \tau}) \to \tau_{ret} & \text{constructors}
\end{array}
$$

$$
\begin{array}{rcll}
d & ::= & \mathsf{in} & \text{copy-in} \\
& | & \mathsf{out} & \text{copy-out} \\
& | & \mathsf{inout} & \text{copy-in-out}
\end{array}
$$

Expressions

$$
\begin{array}{rcll}
exp & ::= & b & \text{booleans} \\
& | & n_w & \text{integers} \\
& | & x & \text{variables} \\
& | & exp_1[exp_2] & \text{array accesses} \\
& | & exp_1[exp_2{:}exp_3] & \text{bitstring slices} \\
& | & \ominus\ exp & \text{unary ops} \\
& | & exp_1 \oplus exp_2 & \text{binary ops} \\
& | & (\tau)\ exp & \text{casts} \\
& | & \{\overline{f = exp}\} & \text{records} \\
& | & exp.f & \text{fields} \\
& | & X.f & \text{type members} \\
& | & exp\langle \overline{\rho} \rangle (\overline{exp}) & \text{function call}
\end{array}
$$

Statements

$$
\begin{array}{rcll}
stmt & ::= & exp\langle \overline{\rho} \rangle (\overline{exp}) & \text{method call} \\
& | & exp := exp & \text{assignment} \\
& | & \mathsf{if}\ (exp)\ stmt\ \mathsf{else}\ stmt & \text{conditional} \\
& | & \{\overline{stmt}\} & \text{sequencing} \\
& | & \mathsf{exit} & \text{exit} \\
& | & \mathsf{return}\ exp & \text{return} \\
& | & var\_decl & \text{variable declaration}
\end{array}
$$

Petr4 Semantics

Type System

$$\frac{\Sigma, \Gamma, \Delta \vdash exp : \text{function}\langle\overline{X}\rangle(\overline{d\ x : \tau}) \rightarrow \tau_{ret} \quad \Sigma, \Delta[\overline{X = \rho}] \vdash \overline{\tau} \rightsquigarrow \overline{\tau'} \quad \Sigma, \Gamma, \Delta \vdash \overline{exp : \tau' \text{ goes } d} \quad \Sigma, \Delta[\overline{X = \rho}] \vdash \tau_{ret} \rightsquigarrow \tau'_{ret}}{\Sigma, \Gamma, \Delta \vdash exp\langle\overline{\rho}\rangle(\overline{exp}) : \tau'_{ret} \text{ goes in}} \text{ T-CALL}$$

Operational Semantics

$$\frac{\begin{array}{c}\langle C, \Delta, \sigma, \epsilon, exp\rangle \Downarrow \langle \sigma_1, \text{clos}(\epsilon_c, \overline{X}, \overline{d\ x : \tau}, \tau, \overline{decl}\ stmt)\rangle \\ \langle \Delta[\overline{X = \rho}], \sigma, \epsilon, \overline{\tau}\rangle \Downarrow_\tau \overline{\tau'} \\ \langle C, \Delta, \sigma_1, \epsilon, \overline{d\ x : \tau' := exp}\rangle \Downarrow_{copy} \langle \sigma_2, \overline{x \mapsto \ell}, \overline{lval := \ell}\rangle \\ \langle C, \Delta[\overline{X = \rho}], \sigma_2, \epsilon_c[\overline{x \mapsto \ell}], \overline{decl}\rangle \Downarrow \langle \Delta_2, \sigma_3, \epsilon_2, \text{continue}\rangle \\ \langle C, \Delta_2, \sigma_3, \epsilon_2, stmt\rangle \Downarrow \langle \sigma_4, \epsilon_3, \text{return } val\rangle \quad \langle C, \Delta, \sigma_4, \epsilon, \overline{lval := \sigma_4(\ell)}\rangle \Downarrow_{write} \sigma_5\end{array}}{\langle C, \Delta, \sigma, \epsilon, exp\langle\overline{\rho}\rangle(\overline{exp})\rangle \Downarrow \langle \sigma_5, val\rangle} \text{ E-CALL}$$

Petr4 Metatheory

Definition 3.1. A statement configuration $\langle C, \Delta, \sigma, \epsilon, stmt \rangle$ is said to be *safe under* $\Xi, \Sigma, \Gamma, \Delta$ *with typing outputs* Σ', Γ', written $\Xi, \Sigma, \Gamma, \Delta \vdash \langle C, \Delta, \sigma, \epsilon, stmt \rangle \dashv \Sigma', \Gamma'$, if (i) $\Xi, \Delta \vdash \sigma$, (ii) $\Xi, \Delta \vdash \epsilon : \Gamma$, and (iii) $\Sigma, \Gamma, \Delta \vdash stmt \dashv \Sigma', \Gamma'$.

Theorem 3.2. *Let $\langle C, \Delta, \sigma, \epsilon, stmt \rangle$ be an initial configuration and take contexts $\Xi, \Sigma, \Gamma, \Delta$. Suppose the configuration is safe under the contexts with typing outputs Σ' and Γ' and $\Sigma \vdash \langle \sigma, \epsilon \rangle$. There exists a final configuration $\langle \sigma', \epsilon', sig \rangle$ and a store typing $\Xi' \supseteq \Xi$ such that $\langle C, \Delta, \sigma, \epsilon, stmt \rangle \Downarrow \langle \sigma', \epsilon', sig \rangle$ and $\Xi', \Sigma', \Gamma', \Delta \vdash \langle \sigma', \epsilon', sig \rangle$.*

Proof is mostly standard, but needs a logical relations argument for termination…

Future Work

- Poulet4: Coq port (with Princeton gang)
- Verified compiler transformations
  - -Inlining functions / controls
  - -Parser unrolling / vectorization
  - -Code motion
- Code generation for new targets
  - -eBPF
  - -FPGAs
  - -Bluespec/Kami/ChiselFlow

End Game

- Final Project (due today; 48-hour extension until Friday)
- Course Evaluations!
- (Regrades)

After 4110

Cornell Courses

- CS 4120 (Compilers)
- CS 5114 / 6114 (Network PL)
- CS 6110 (Advanced PL)
- CS 6120 (Advanced Compilers)
- CS 61xx (Special Topics)

Research

- CS 4999
- Summer
- Open Source (e.g., GSOC)

Doing a PhD…

Applications

- Transcript
- Recommendation Letters
- Statements
- (GRE)

Masters Degree?

- Optional not required (in North America)

Conferences

- SIGPLAN "Big 4": POPL, PLDI, ICFP, OOPSLA
- Programming Languages Mentoring Workshop (PLMW)

Industry

Compilers

- GPUs
- TPUs
- LLVM
- WebAssembly
- Rust

Verification

- Startups (Bedrock Systems, Correct Computation, etc.)
- Amazon Automated Reasoning Group
- Google Project Oak

🙏 Thank you!

- CS 4110 is one of my favorite classes to teach…
- Hybrid classes are hard for all of us…
- Thanks for your enthusiasm this semester, for engaging in Zoom, and for patience with my tech SNAFUs
- Please keep in touch!